FREE TO ENTER COMPETITION SINCLAIR SOFTWARE TO BE WON ££££'S

AN ARGUS SPECIALIST PUBLICATION

MACHINE CODE PT 5 91

SPECTRAMON PT2 28

ZX COMPUTING

June/July '83

$3.75

£1.85

Britain's Biggest Magazine For The Sinclair User



Over 120 pages of information and programs for the ZX Spectrum, ZX81 and ZX80 computers

NEW FEATURE
YOUR REVIEWS PUBLISHED!

- Arcade action on your ZX81 and Spectrum
- Programming problems? Help is at hand
- Business and domestic routines for your micro
- Educational programs – six of the best!

ZX COM

ZX Computing
Volume One
Number Seven
June/July 1983

Editor: Roger Munford
Advertising Manager: Miriam Roberts
Managing Editor: Ron Harris
Managing Director: T J Connell

Origination and design by MM Design & Print, 145 Charing Cross Road, London WC2H 0EE.

Published by Argus Specialist Publications Ltd, 145 Charing Cross Road, London WC2H 0EE.

CONTENTS



Hewson Assoc.

MPUTING

ZX Computing is published bi-monthly on the fourth Friday of the month. Distributed by: Argus Press Sales & Distribution Ltd, 12-18 Paul Street, London EC2A 4JS. 01-247 8233. Printed by: Henry Garnett Ltd., Rotherham.



Paramount Pictures

ZX Computing is constantly on the look-out for well-written articles and programs. If you think that your efforts meet our standards, please feel free to submit your work to us for consideration.

All submitted material should be typed if possible; handwritten work will be considered, but please use your neatest handwriting. Any programs submitted should be listed, a cassette of your program alone will not be considered. All programs must come complete with a full explanation of the operation and, where relevant, the structure; Spectrum programs should be accompanied with a cassette of the program (which will be returned) as well as the listing.

All submissions will be acknowledged and any published work will be paid for at competitive rates. All work for consideration should be sent to the Editor at our Charing Cross Road address.

Welcome



Greetings, and welcome to this our seventh issue of *ZX Computing*. Within these pages you'll be assailed with a galaxy of games programs for your ZX Spectrum, ZX81 and ZX80, as well as a good selection of business, educational and domestic software for you to RUN.

The state of play

I don't suppose I need tell you just how successful Sinclair Research have been selling their micros throughout the world. But the news that they have sold over one million computers is certainly a great achievement in the computer marketplace.

In addition to this figure, Sinclair Research also claim that around 600,000 computers have been manufactured under licence by Timex for the North American market. So, you can be sure of one thing — as a Sinclair user, you are not alone!

On a more important note, I hope all of you have read of the warning which Sinclair Research announced concerning their faulty Spectrum power supplies. Unfortunately, the warning was only announced days after *ZX Computing's* press day, so I was unable to include it as part of the news section. However, their warning was given immediate and wide coverage in the computer press and according to the people at Sinclair has brought a massive response.

Before anyone out there panics, the Spectrums involved were only those bought between January 1 and the end of February, and then only if the lead between the power supply and the computer was black with a white strip. Also, it only affected power supplies with a smooth surfaced (rather than textured) finger grip on the jack plug.

If you're in any doubt at all about your power supply, you should get in contact with the team down at Sinclair Research Ltd, Unit F, Broad Lane, Cottenham, Cambridgeshire CB4 4SW.

Issuing forth

Amongst the 'goodies' in this issue are a selection of great games for your Spectrum, ZX81 and ZX80. Just as the standard of arcade games has improved by leaps and bounds, so has the standard of software submitted to our offices.

Many of you are using the arcades for inspiration in your programming and I am pleased to include a number of programs which you may recognize. There is Leprechaun's gold, an exciting maze game in which you have to hunt out a pot of gold, and Muncher, in which you are trying to eat the dots and escape the monsters. You may also like to try the program, Every pitcher tells a story, a baseball game (sorry about the play on words). You'll find a whole lot more in this issue, but I'll leave the rest for you to find.

Within these pages, you'll also be regailed with a number of reviews by our sterling review team. James Walsh may have bitten off more than he can chew, but he makes a valiant attempt to review 12 (count 'em) software packages for your ZX Spectrum. Also, Nick Pearce takes a look at some of the latest software for the ZX81.

And featuring . . .

Two special features begin this issue, one of which I hope will help you in your programming efforts, the other hopefully providing a showcase of your thoughts about the commercial software you buy.

Problem page will be a regular spot for you to ask Peter Shaw, author of 'Games for your Spectrum', for help with your programming. If you would like to ask Peter for any advice concerning software or hardware for your Sinclair computer, please try to include as much detail as you can concerning your problem.

Also, beginning this issue is a feature called Reader's reviews. This is where you come in. All you have to do is to write a review of any software you buy, including details of manufacturer, price, etc, and send it to us. If it gets published, we'll pay for the software package you reviewed. So, if your review gets published, not only do you see your opinions in print, but you get your software free too!

Contributions

We are always on the lookout for good programs and articles for future issues of *ZX Computing*, and where better to look than to our own readers. If, when reading through the magazine, you think you can write programs as well, or better than, our present contributors, then let's hear from you.

All contributions are, of course, paid for at very competitive rates. So if you've got your eye on a new ZX add-on or you'd just like to supplement your pocket money, get writing! It is vital, though, that all the programs you send us are totally original, and not 'borrowed' or 'adapted' from other magazines or books. (When Tim Hartnell was sitting in the Editor's chair, he even received 'original' contributions he himself had written for his own books!)

Any kind of program (business, domestic, educational, or just fun) will be welcomed, but particularly those which use ZX BASIC in clever and efficient ways, or those which employ certain routines which can be re-used in other programs.

Program listings are vital, along with a clear explanation of how the program is constructed, what it does and what the user can expect to see once the program is RUN (a screen dump is particularly valuable in this respect). When submitting Spectrum programs, it is very important to remember to enclose a cassette of the program as well as the listing, as this will allow us to check the program before publication.

On a final note

Well, all I can say is I hope you have a good time and use this magazine to the full. So, settle down at your computer, open up the pages, and get down to the serious business of making the most of your micro with *ZX Computing*.

Roger Munford.



A touch of genius

Dear ZX Computing,
I have at last got around to typing out the utility programs contained in the article, 'Scrolling that screen window', in the Feb/Mar issue of *ZX Computing*.

I have actually penned an additional piece of code which you can see below. I have no doubt that a similar effect can be produced by a shorter program and I look forward to seing if any of your other readers can write it.

The program below is an adaptation of the first listing and defines a window and then inverts the window, but not the rest of the screen. This would be useful if, say, you did not want to invert a border during a game.

```
2A OC 40     Start
23
OE (13)
3E (OF)      Rectangle
B9
30 06
11 21 00
19
18 14
06 20        Begin
3E (1C)      Coloumn
B8
38 09
3E (04)
B8
30 04
7E
C6 80
77
23           Next
10 EF
23
OD           End
20 DE
C9
```

I have not included a narrative because comparison with Listing 1 in the article will show how it works. The code in brackets is variable and sets the window dimensions, the same as Listing 1.
Yours faithfully,

John Schofield,
Blackburn.



Strange . . .

Dear ZX Computing,
I was typing in a few RAND USR statements on my ZX81 with 16K RAM Pack for fun, when I suddenly came across some strange codes. Here are a few for you to try.

Try typing in 'RAND USR 200'. If you look at the report code it states 'CHR$ 136/50. The most weird code I found was 'H'. The statements I typed in to get this code were 'RAND USR 5900' and 'RAND USR 5904'. Two things happen when you type these in.

The first result is that three CHR$ 1s form a triangle in the middle of the screen and no report code is given. The bottom two lines of the screen blanked out and a further command caused the dots to scroll down the screen and then a system crash occured.

The second result was that the three CHR$ 1s were printed in a diagonal on the right-hand side of the screen and H/O was given as a report code. However, if Newline was pressed, the dots scrolled down the screen and eventually, a '0' and 29 inverse '>'s can be seen, but then the system crashes.

Another report code I found was an inverse asterisk/0, which I discovered by POKEing USR 5900, any number.

I wonder if I am the first person to notice these codes and if there are any more yet to be discovered.
Yours faithfully,

Toby Philpott,
Folkestone.



Not 'alf

Dear ZX Computing,
With reference to M Clayton's letter in the Feb/March issue of *ZX Computing* regarding the mysterious '0.5' which could not be trapped, I have also come across the same problem. My program had the task of converting fractions of minutes into seconds which 'hung up' on any multiple of five (as I later found out), for example, 0.5 or fractions adding up to, say 500.

As a computer user of some years practice, I simply couldn't rationalise a problem such as this being allowed to become 'acceptable'.

The following solution works for the problem as stated, although I don't like the implications!

```
 5 LET B = ½
10 LET A = 256
20 LET A = A/2
30 PRINT, A
40 IF A = B THEN STOP
50 GOTO 20
```

I have also typed in the excellent 'Scrolling that screen window' which appeared in the same issue, written by J Elliot. The machine code works perfectly, but only when the amendments/ additions shown in Fig. 1 are made to Listing 4. By way of explanation, the address of the routines is critical to the re-definition (and re-re-definition) of both 'window' and 'scroll' — as seen in the demonstration program, Listing 5. (Lines 360 and 400 use the base address con-offset.) However, in Listing 4, as soon as line 400 is executed, the address contained in B is corrupted by the addition to it of the offset, since the result is placed back into B.
Yours faithfully,

DR Williams,
Rhondda,
Mid-Glamorgan.

45 LET BB = B	Save start of routine address.
280 POKE B + 25,32 − H	Error correction.
290 POKE B + 30 + L, CODE K$	Error correction.
295 LET B = BB	Restore start of routine address.
600 LET Q = USR(B + 94)	Demonstrate screen inverse.
605 PAUSE 100	Renumber only.

Fig. 1.

Half cocked?

Dear ZX Computing,
I have been having similar problems to M Clayton judging by his letter in the Feb/Mar issue of *ZX Computing*.

The solution does, of course, work, and the reason 'remember that if A is not exactly equal to 0.5 in line 40' is, of course, valid. But which one is wrong, A or 0.5?

The assumption most of us up till now have been making is that '0.5 equals a half'; after all, we were all taught that at school — it seems reasonable so why not believe it? Because on the ZX81 it is not true!

Examination of the bit patterns show that A is exactly equal to a half, but that (shock horror!) 0.5 is

not! A is stored as five bytes: 128,0,0,0,0; which decodes to $2^{-1}+0+0+0+0$ — which is exactly a half. Whereas if you look at '0.5' you can see it is also stored in five bytes: 127, 127, 255, 255, 255; which decodes to $2^{-2}+2^{-3}+2^{-4}$, etc, until 2^{-33}, which is less than a half.

If you amend line 40 to:

```
40 IF A=0.5+2**-33 THEN
   STOP
```

then the storage error is corrected and all is well.

On a more practical note, the use of reciprocals, powers, multiplication or division to avoid the troublesome value usually works wonders.

For example:

```
IF A=½
IF A=2**-1
IF A*2=1
IF A*1001=500.5
IF A/10=0.5
```

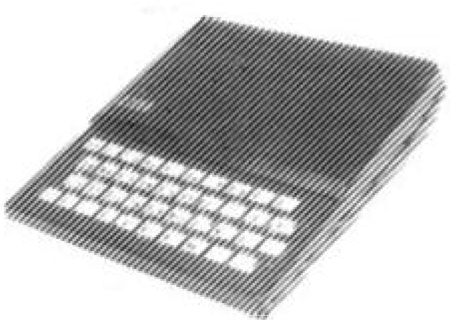
All of the above work; in fact, almost anything works as long as you avoid the dreaded variable, 0.5. Paulo Xavier's use of $ and STR$ also solves M Clayton's problem, though his own problem is caused by something else — but that's another long, long story.

The following notes might also be useful to other readers. Many square roots are not exact even when they should be integers. Paulo had about one chance in three of picking on a dud. Also, an awful lot of numbers do not like being divided by the number, 1.6. However, my own study has shown that many, if not all, of these are predictable.

The whole field is a rich source of program bugs; and my advice to anyone wary enough to take the challenge is to learn to read floating point binary and look at the bit patterns. This should help you avoid the troublesome values.

Yours faithfully,

HM Tucker,
Luton,
Bedforshire.



Pretty smooth!

Dear ZX Computing,
I wonder if any of your readers would be interested in this machine code routine which I have written for the ZX Spectrum.

This program LOADs the machine code routine in the DATA statement, and then SAVEs it for future use as a subroutine in any BASIC program. The routine itself, when called, will scroll the whole screen up one pixel. Thus, to scroll one line would require a FOR. . . NEXT of '8'.

When the program is RUN, it asks for the start address where the code is to be POKEd. This can be anywhere in the free RAM, but is most likely to be 32500 in a 16K Spectrum and 65200 in a 48K machine.

To call the routine shown in Fig. 2, use 'RANDOMIZE USER s' (s being the value which you input when the program asked for the start address).

Yours faithfully,

Paul Maycock,
Bristol.

```
 100 >REM    By P Maycock  18/1/83
 110 DATA 33,0,65,17,0,64,6,3,19
7,6,8,197,6,7,197,1,32,0,237,176
,1,224,0,9,229,213,225,9,229,209
,225,193,16,236,1,224,7,237,66,1
,32,0,237,176,1,32,0,237,66,229
 120 DATA 209,1,0,1,9,193,16,209
,1,0,7,213,225,9,1,32,0,229,237,
65,229,209,225,1,32,0,237,176,1,
0,1,213,225,9,193,16,177,33,224,
87,1,0,32,113,35,16,252,201
 130 INPUT "Start address ?",s
 140 FOR f=0 TO 97
 150 READ a
 160 POKE s+f,a
 170 NEXT f
 190 SAVE "scroll mc"CODE s,98
 200 INPUT "Press ENTER to verif
y"; LINE a$
 210 VERIFY "scroll mc"CODE
```

Fig. 2.

Read all about it . . .

Dear ZX Computing,
Mr Pulsford, in his article in the Feb/Mar issue of *ZX Computing*, has re-discovered approximations for the number of primes in a given interval. But better approximations have been known for nearly two centuries as can be found in a number of authorative books on the subject.

I append three easily accessible references for your readers (shown in Fig. 3.), LeVeque being the most recent (and containing and extensive bibliography), Grosswald providing a slightly different and very readable account about the distribution of prime numbers, and Beiler (my personal favourite) which has recently been re-printed.

Yours faithfully,

Ed Rosentiel, MDS, BSc.
Emeritus Lecturer,
King's College Hospital
Medical School,
London.

Beiler, AH	Recreations in the Theory of Numbers — The Queen of Mathematics Entertains, Dover Publications Inc., New York.
LeVeque, WJ	Fundamentals of Number Theory, Addison Wesley Publishing Company.
Grosswald, E	Topics from the Theory of Numbers, The Macmillan Company, New York.

Fig. 3.

ZX mirage?

Dear ZX Computing,
A simple question for you. Is the ZX Microdrive a mirage? Ever since the ZX Spectrum was first advertised, we have been promised that 'the Microdrives will be available later this year'. This statement seems to have a hint of 'tomorrow never comes' or should I say 'later this year' never comes.

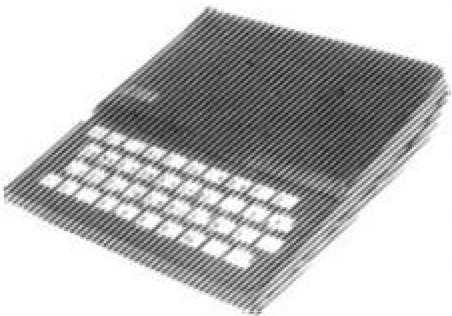
What are Sinclair Researcb doing? They seem to know its specifications; after all, there has been a picture in every advertisement for the ZX Spectrum. Are they stock-piling Microdrives?

Whatever the reason, please hurry up with the Microdrives.

Yours faithfully,

Robert Aykroyd,
Leeds.

● *There are one or two of you out there with the same thoughts going through your head I've no doubt. So, I had a word with a Sinclair spokesperson who assured me that the Microdrives will be here in the 'very, very near future'. Apparently, the original design was superseded by an improved one, thus making the introduction date a little late. However, since the design was frozen late last year, everything has gone according to schedule and Sinclair Research are confident that the ZX Microdrives will soon appear. As a side note, the Sinclair spokesperson added that when the Microdrives do appear they will not be immediately available to everyone, but those who bought their Spectrums first will be offered the chance to buy Microdrives first — this, they feel, will give the people who have waited the longest 'a fair crack at the whip'.*



In reply . . .

Dear ZX Computing,
Re J Crawford's letter in the Feb/Mar issue of *ZX Computing*, when I ran the three line program I got a blank screen and recovered the listing by pressing Newline.

However, by erasing line 30, the program PRINTs whatever you like at 23,5. One may also PRINT at 22,5 if you wish.

Yours faithfully,

MJ O'Regan,
Geneva.
Switzerland.



Build up your display

Dear ZX Computing,
While experimenting with my ZX81's display, I hit upon a very useful routine to expand the display file to 34 columns by 24 lines, rather than the usual 32 by 24.

This is all very good, but the catch is that on the ZX81 with more than 3¼K RAM, the pro-



gram causes a system crash. Nevertheless, it can be used with great effect to add a little more action to games, etc. Here is the program:

```
10 POKE 16441,34
20 PRINT "34 CHARACTERS"
```

I hope this will be of use to people wanting that extra bit of space in their programs.
Yours faithfully,

J Sahota,
Hampshire.



Bug bound

Dear ZX Computing,
I am owner of a recently updated 16K ZX81. However, for several months, I have had to make do with 1K. I therefore understand the problems and frutrations of 1K owners and I was very interested to see the program, Home run, in the Feb/Mar issue of *ZX Computing*.

I was very impressed with the program, although one or two bugs did manage to creep in. The '6' and '7' in lines 110 and 120 should be swapped around so as to match the directions of the arrows on the cursor keys. Also, line 220 should have read 'GOTO 20' instead of 'GOTO 10' (which would reset C to 50 and thus lose the effect of the decrement in line 200).

Despite these minor changes, I found it a very enjoyable and addictive program.
Yours faithfully,

Mark Armstrong,
Lincoln.



Bubble-headed?

Dear ZX Computing,
M Biddell's attempt 'to make sense of bubble sorting' in the Dec/Jan issue of ZX Computing seems to use very complex coding to do a task which is simplicity itself — a classic case of 'bubble-headed' thinking perhaps!

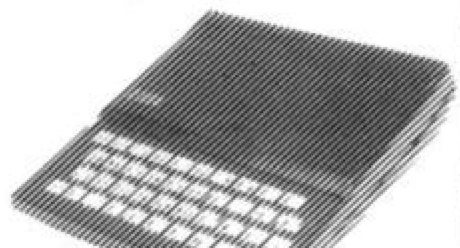
All you need are two subscripts, S (senior) and J (junior), and the main processing can then be done in just a few lines:

```
50 FOR J = 1 TO 8
60 FOR S = 8 TO (J + 1) STEP − 1
70 IF A(S) < A(J) THEN
   LET M = A(S): LET
   A(S) = A(J): LET
   A(J) = M
90 NEXT S
95 NEXT J
```

Mike's other variables, I, K and B, are superflous as you can just PRINT A(J) instead of B.

The other advantage of the above simplified code is that it's much easier to understand what's happening.
Yours faithfully,

Alan Lawson,
Edinburgh.



Fade to grey?

Dear ZX Computing,
When I first bought my ZX Spectrum, I didn't realise that I would use it to the extent that I have. This causes considerable wear on the keyboard and the printing on the keys began to fade.

However, I have solved that problem by taping a nine centimetre by 24 cm piece of cling film over the ZX Spectrum's keyboard. When this becomes worn and tattered, I just peel it off and apply a fresh sheet thus leaving an untouched keyboard underneath.
Yours faithfully,

NC Felgate,
Plymouth.



A helping hand . .

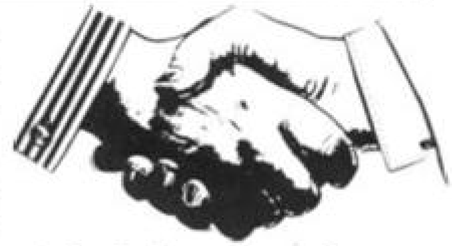
Dear ZX Computing,
I have heard so much about so many people having problems LOADing programs from tape into the ZX81. I have had my ZX81 for nearly a year now, but six months ago I began encountering the dreaded LOADing problems. There were occasions when the TV screen went blank and the cursor refused to appear, and there were times when the computer stopped LOADing suddenly, although this time the cursor re-appeared.

The number of times this happened began to increase, but after close inspection I found it was the tape recorder that needed 'doctoring'. Below, I have provided three methods for dealing with the problems I was encountering.

1 — Try adjusting the volume level so that the thick horizontal lines displayed when a program is being LOADed are approximately twice the size of the spaces in between. But beware, changing the volume too drastically will cause the computer to stop LOADing.
2 — Try recording programs with only the two MIC plugs connected, and LOADing with only the two EAR plugs in.
3 — If the cassette player does not seem to be producing as loud a sound as it should, then use a cassette head cleaner. I found that this, used in conjunction with dust and oxide remover, was very beneficial.

All three methods have improved my LOADing success a lot and I hope that they will work for you too.
Yours faithfully,

Malcolm Boyd,
Belfast.



MAIL ORDER PROTECTION SCHEME

If you order goods from Mail Order Advertisers in this magazine and pay by post in advance of delivery, this publication will consider you for compensation if the advertiser should become insolvent or bankrupt, provided:

1. You have not received the goods or had your money returned; and
2. You write to the publisher of this publication explaining the position not earlier than 28 days from the day you sent your order and not later than 2 months from that day.

Please do not wait until the last moment to inform us. When you write, we will tell you how to make your claim and what evidence of payment is required.

We guarantee to meet claims from readers made in accordance with the above procedure as soon as possible after the advertiser has been declared bankrupt or insolvent to a limit of £1,800 per annum for any one advertiser, so affected, and up to £5,400 p.a. in respect of all insolvent advertisers. Claims may be paid for higher amounts, or when the above procedures have not been complied with, at the discretion of this publication, but we do not guarantee to do so in view of the need to set some limit to this commitment and to learn quickly of reader's difficulties.

This guarantee covers only advance payment sent in direct response to an advertisement in this magazine (not, for example, payments made in response to catalogues, etc, received as a result of answering such advertisements):

CLASSIFIED ADVERTISEMENTS ARE EXCLUDED.

Muncher

A Spectrum version of the popular arcade game from Robert Turner of Cwmbran, Gwent.



This game involves your character (which looks like a heart on its side) travelling around the maze eating dots. You are pursued by two monsters, which if they catch you will devour you. Luckily, in the corners of the maze are power pills which, if eaten by your character, will enable you to chase the monsters and kill them. But you'll have to hurry because the pills only last a limited period of time after which the monsters are after you again. You'll know when the monsters are vulnerable because the monsters turn green and begin flashing.

Sometimes a cherry will appear under the monster's den and this is worth 10 points. Each dot you eat is worth one point each, and for each power pill you eat and each monster you kill you get 10 points each. There's lots of opportunity to gain a high score — all you have to do is to stay alive long enough!

Variables

The variables used in the program are:

S	— Score.
HS	— High score.
T	— Score when the screen is cleared.
B$	— Where the mazed is stored.
COUNT	— If this is less than 30, the monsters can be eaten. If this amount is more than 30 the monsters can eat you.
LIVES	— The number of lives you have left.
Y and X	— The monster's position.
A$	— The shape of the monster.
GX and GY	— The position of the first monster.
GX1 and GY1	— The position of the second monster.
C	— The movement of the monsters. If C = −1, then the monsters move away from you, if C = 1 then they're after you.
D$	— The shape under monster one.
E$	— The shape under monster two.
A	— Skill level.
N	— Allotted for general use.

Program description

Lines	Description
1-2	Set up the user definable graphics.
3-4	Set up some of the variables.
10-50	The data for the graphics.
60-260	Set up the maze in B$.
270	PRINTs the maze on the screen.
280-290	Set up some more of the variables.
295	PRINTs the highest score so far.
300-320	PRINT the monsters and your character.

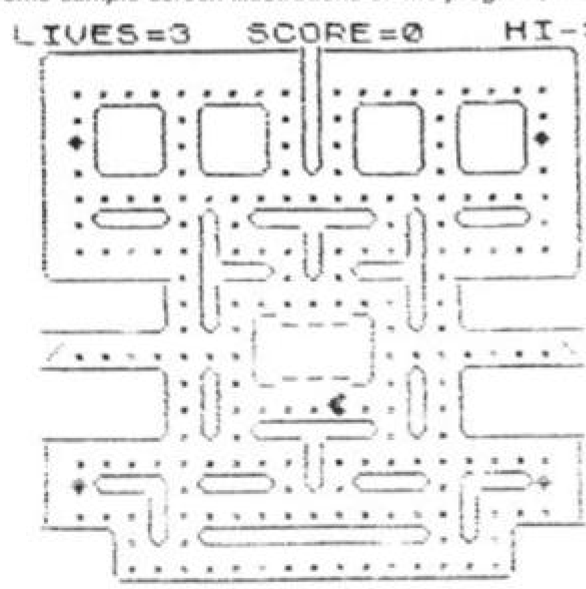


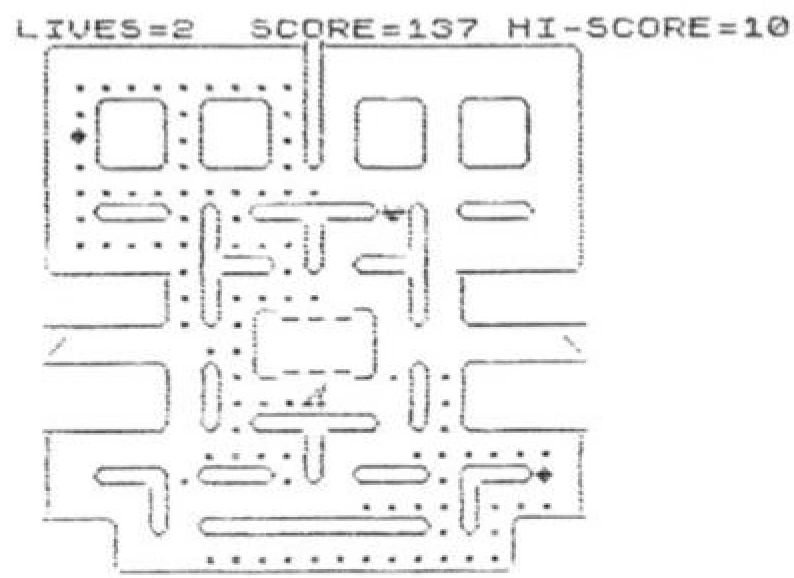
330	Decides whether or not to PRINT the cherry.
340-370	Scan the keyboard and goes to the corresponding line number.
375	PRINTs score.
380-400	See what's at the monster's position and increases the score to the corresponding level.
410	PRINTs your character on the screen.
415	If the screen is cleared, GOes TO line 5000.
420	PRINTs things under the monsters.
425-490	Find positions of monsters and moves it accordingly.
500	Increases the COUNT.
510-535	PRINT the monsters. Flashing green if the COUNT is less than 30. Cyan and magenta are used if the COUNT is more than 30.
540-545	If your character and the monster share the same position, then the program GOes TO line 9000.
550	Updates D$ and E$.
1000-1030	PRINT the cherry.
1500-1540	Move your character to the right.
1600-1640	Move your character to the left.
1700-1730	Move your character up.
1800-1830	Move your character down.
2000-2050	If your character should eat one of the power pills, the variables are re-set accordingly.
5000-5070	Between the screen routine.
7000-7090	Select the skill level.
8010-9000	Your character kills one of the monsters.
9000-9030	When one of your characters is eaten, the variables are updated.
9500-9530	The game is over and the variables are re-set for a new game.

```
   1 RESTORE 0: FOR a=144 TO 164
   2 FOR b=0 TO 7: READ c: POKE
USR CHR$ a+b,c: NEXT b: NEXT a
   3 LET s=0: LET hs=0: LET T=18
0: DIM B$(21,21): LET COUNT=30
   4 LET lives=3
   5 GO SUB 7000
  10 DATA 0,255,0,0,0,0,0,0,0,0,
0,0,0,0,255,0,0,255,0,0,0,0,255,
0,0,248,4,2,2,2,2,2
  20 DATA 0,31,32,64,64,64,64,64
,0,63,64,128,128,64,63,0,0,252,2
,1,1,2,252,0
  30 DATA 2,2,2,2,2,4,248,0,64,6
4,64,64,64,32,31,0,2,2,2,2,2,2,2
,2
  40 DATA 0,24,36,66,66,66,66,66
,66,66,66,66,66,36,24,0,64,64,64
,64,64,64,64,64,66,66,66,66,66,6
6,66,66,0,0,16,56,124,56,16,0
  50 DATA 24,60,30,15,30,60,24,0
,0,0,34,119,127,62,28,8,0,24,60,
120,240,120,60,24,16,56,124,254,
238,66,0,0,56,126,90,219,255,255
,255,147,2,6,10,20,36,68,238,66
  55 PAPER 1: BORDER 1: CLS : IN
K 7: PAPER 0
  60 LET B$(1)="EAAAAAAAAANAAAAA
AAAAD"
  70 LET B$(2)="M.........N.....
....J"
  80 LET B$(3)="M.EAD.EAD.N.EAD.
EAD.J"
  90 LET B$(4)="MOM J.M J.N.M J.
M JOJ"
 100 LET B$(5)="M.IBH.IBH.L.IBH.
IBH.J"
 110 LET B$(6)="M...............
....J"
 120 LET B$(7)="M.FCG.K.FCACG.K.
FCG.J"
 130 LET B$(8)="M.....N...N...N.
....J"
 140 LET B$(9)="IBBBB.MCG.L.FCJ.
BBBBH"
 150 LET B$(10)="    J.N.......N
.M    "
 160 LET B$(11)="BBBBH.L.E---D.L
.IBBBB"
 170 LET B$(12)="/........M   J..
.....\"
 180 LET B$(13)="AAAAD.K.I---H.K
.EAAAA"
 190 LET B$(14)="    J.N.... ..N
.M    "
 200 LET B$(15)="BBBBH.L.FCACG.L
.IBBBB"
 210 LET B$(16)="M.........N....
.....J"
 220 LET B$(17)="MOFCD.FCG.L.FCG
.ECGOJ"
 230 LET B$(18)="M...N..........
.N...J"
 240 LET B$(19)="AAD.L.FCCCCCCCG
.L.EAA"
 250 LET B$(20)="  J............
...M  "
 260 LET B$(21)="   AAAAAAAAAAAA
AAA   "
 270 FOR N=1 TO 21: PRINT AT N,1
;B$(N): BEEP .05,N+15: NEXT N
 280 LET y=14: LET x=12: LET a$=
"R"
 290 LET gx=12: LET gy=12: LET g
x1=12: LET gy1=11: LET C=1: LET
D$=" ": LET E$=" "
 295 PRINT AT 0,19;"HI-SCORE=";h
s
 300 PRINT AT gx,gy; INK 3;"T"
 310 PRINT AT gx1,gy1; INK 5;"T"
 320 PRINT AT y,x; INK 6;a$
 330 IF INT (RND*100)=0 THEN GO
SUB 1000
 340 IF INKEY$="8" THEN GO SUB 1
500
 350 IF INKEY$="5" THEN GO SUB 1
600
 360 IF INKEY$="7" THEN GO SUB 1
700
 370 IF INKEY$="6" THEN GO SUB 1
800
 380 IF B$(Y,X)="." THEN LET s=s
+1: BEEP .05,25
 390 IF B$(Y,X)="O" THEN LET s=s
+10: BEEP .1,25: BEEP .1,39: GO
SUB 2000
 400 IF B$(Y,X)="U" THEN LET s=s
+10: LET T=T+10: BEEP .1,28: BEE
P .1,36
 401 PRINT AT 0,0;"LIVES=";lives
;"   SCORE=";S
 405 LET B$(Y,X)=" "
 410 PRINT AT y,x; INK 6;a$
 415 IF S=T THEN GO TO 5000
 420 PRINT AT GX,GY;D$;AT GX1,GY
1;E$
 425 IF RND<a THEN GO TO 440
 430 IF gy<x THEN LET d=CODE b$(
gx,gy+c): IF d<>47 AND d<>92 AND
 d<144 OR d>157 AND d<>47 AND d<
>92 THEN LET gy=gy+c: GO TO 470
 435 IF RND<a THEN GO TO 450
 440 IF gy>x THEN LET d=CODE b$(
gx,gy-c): IF d<>47 AND d<>92 AND
 d<144 OR d>157 AND d<>47 AND d<
>92 THEN LET gy=gy-c: GO TO 470
 445 IF RND<a THEN GO TO 460
 450 IF gx<y THEN LET d=CODE b$(
gx+c,gy): IF d<>47 AND d<>92 AND
 d<144 OR d>157 AND d<>47 AND d<
>92 THEN LET gx=gx+c: GO TO 470
 455 IF RND<a THEN GO TO 470
 460 IF gx>y THEN LET d=CODE b$(
gx-c,gy): IF d<>47 AND d<>92 AND
 d<144 OR d>157 AND d<>47 AND d<
>92 THEN LET gx=gx-c
 465 IF RND<a THEN GO TO 475
 470 IF gy1<x THEN LET d=CODE b$
(gx1,gy1+c): IF d<>47 AND d<>92
AND (d<144 OR d>157) THEN LET gy
1=gy1+c: GO TO 500
 472 IF RND<a THEN GO TO 480
 475 IF gy1>x THEN LET d=CODE b$
(gx1,gy1-c): IF d<>47 AND d<>92
AND d<144 OR d>157 AND d<>47 AND
 d<>92 THEN LET gy1=gy1-c: GO TO
 500
 477 IF RND<a THEN GO TO 490
 480 IF gx1<y THEN LET d=CODE b$
(gx1+c,gy1): IF d<>47 AND d<>92
AND d<144 OR d>157 AND d<>47 AND
 d<>92 THEN LET gx1=gx1+c: GO TO
 500
 485 IF RND<a THEN GO TO 500
 490 IF gx1>y THEN LET d=CODE b$
(gx1-c,gy1): IF d<>47 AND d<>92
AND d<144 OR d>157 AND d<>47 AND
 d<>92 THEN LET gx1=gx1-c
 500 LET COUNT=COUNT+1
 510 IF COUNT>=30 THEN LET C=1
 520 IF C=-1 THEN PRINT AT GX,GY
; INK 4; FLASH 1;"T"
 525 IF C=1 THEN PRINT AT GX,GY;
 INK 3;"T"
 530 IF C=-1 THEN PRINT AT GX1,G
Y1; INK 4; FLASH 1;"T"
 535 IF C=1 THEN PRINT AT GX1,GY
1; INK 5;"T"
 540 IF GX=y AND GY=x THEN GO TO
 9000
 545 IF GX1=y AND GY1=x THEN GO
TO 9000
 550 FLASH 0: LET D$=B$(GX,GY):
LET E$=B$(GX1,GY1): GO TO 330
1000 IF B$(14,11)="U" THEN RETUR
N
1010 PRINT AT 14,11; INK 2;"U":
BEEP .5,15: IF B$(14,11)="." THE
N LET T=T-1
1020 LET B$(14,11)="U"
```

```
1030 RETURN
1500 PRINT AT Y,X;" "
1505 IF B$(y,x+1)="." THEN LET x=x+1: GO TO 1530
1510 IF B$(y,x+1)=" " THEN LET x=x+1: GO TO 1530
1520 IF B$(y,x+1)="U" THEN LET x=x+1
1525 IF B$(y,x+1)="\" THEN LET x=2
1530 LET a$="R"
1540 RETURN
1600 PRINT AT Y,X;" "
1605 IF B$(y,x-1)="." THEN LET x=X-1: GO TO 1630
1610 IF B$(y,x-1)=" " THEN LET x=X-1: GO TO 1630
1615 IF B$(y,x-1)="O" THEN LET x=X-1: GO TO 1630
1620 IF B$(y,x-1)="/" THEN LET x=20: GO TO 1630
1625 IF B$(y,x-1)="U" THEN LET x=X-1
1630 LET a$="P"
1640 RETURN
1700 PRINT AT Y,X;" "
1705 IF B$(y-1,x)="." THEN LET y=y-1: GO TO 1725
1710 IF B$(y-1,x)="O" THEN LET y=y-1: GO TO 1725
1720 IF B$(y-1,x)=" " THEN LET y=y-1
1725 LET a$="Q"
1730 RETURN
1800 PRINT AT y,x;" "
1805 IF B$(y+1,x)="." THEN LET y=y+1: GO TO 1825
1810 IF B$(y+1,x)="O" THEN LET y=y+1: GO TO 1825
1815 IF B$(y+1,x)=" " THEN LET y=y+1
1825 LET a$="S"
1830 RETURN
2000 LET C=-1
2010 LET B$(Y,X)=" "
2020 PRINT AT Y,X;A$
2030 LET T=T+10
2040 LET COUNT=0
2050 RETURN
3000 IF Y>D THEN LET D=D+G
3010 IF X>F THEN LET F=F+H
3020 IF X<F THEN LET F=F-H
3030 IF Y<D THEN LET D=D-G
3040 RETURN
4010 LET D=GX: LET F=GY: RETURN
4020 LET D=GX1: LET F=GY1
4030 RETURN
5000 CLS : PRINT AT 10,31;"O"
5010 FOR N=1 TO 27
5020 PRINT AT 10,N; INK 5;" T " INK 7;"R"
5030 BEEP .05,N: NEXT N
5040 FOR N=27 TO 1 STEP -1: PRINT AT 10,N; FLASH 1; INK 4;"T"; FLASH 0; INK 7;" P "
5050 BEEP .05,N: NEXT N
5060 LET T=T+180
5070 GO TO 60
6010 LET GX=D: LET GY=F: RETURN
6020 LET GX1=D: LET GY1=F: RETURN
7000>PRINT AT 0,12;"CHOMPER";TAB 11;"█████"
7010 PRINT AT 9,0;"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA"
7020 PRINT AT 11,3;"SELECT SKILL LEVEL (1-5) "
7030 PRINT TAB 7;"(5 is the easiest)"
7040 PRINT AT 14,0;"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA"
7060 LET a=CODE INKEY$-49
7070 IF a>5 OR a<0 THEN GO TO 7060
7080 LET a=a+1: LET a=a/10
7090 RETURN
6000 IF gy=x AND gx=y THEN LET s=s+10: LET t=t+10: LET gy=12: LET gx=12: BEEP .05,20: BEEP .05,10: BEEP .1,10: GO TO 550
8010 IF gy1=x AND gx1=y THEN LET s=s+10: LET t=t+10: LET gy1=11: LET gx1=12: BEEP .05,20: BEEP .05,10: BEEP .1,10: GO TO 550
9000 IF count<=30 THEN GO TO 8000
9005 LET LIVES=LIVES-1
9010 PRINT AT Y,X; FLASH 1;A$: FOR N=50 TO 0 STEP -1: BEEP .05,N: NEXT N
9020 IF lives<=0 THEN GO TO 9500
9030 GO TO 270
9500 IF HS<S THEN LET HS=S
9510 PRINT AT 10,0;"HIT ANY KEY TO RESTART"
9520 IF INKEY$="" THEN GO TO 9520
9530 PAPER 1: LET S=0: CLS : PAPER 0: LET T=180: LET lives=3: GO TO 60
```

Some sample screen illustrations of the program, Muncher.





A breakdown of the user-defined graphics used in the program and to which graphics key they are assigned.

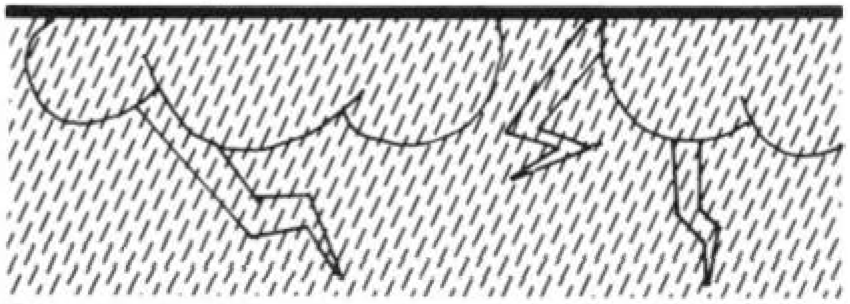
A=‾ B=_ C=⊏ D=⌐ E=┌ F=⊂ G=⊃ H=┘

I=└ J=| K=∩ L=∪ M=| N=|| O=◆ P=❯

Q=♥ R=❮ S=▲ T=👾 U=⩘

Weather report

See if you can predict the next bout of stormy weather with this excellent program written for us by Cathryn Corns of Enfield.





This program will plot out histograms of weather data — maximum temperature, minimum temperature, humidity, barometric pressure, hours of sun and inches of rain, for any calendar month. You do not need to collect the data yourself — most daily papers carry at least some of the information.

The program has proved useful at home and in the classroom. The built-in error correction routine allows wrongly entered results to be corrected simply; this has proved particularly useful when children use the program. There is also a facility provided to allow for printing out of data previously entered. This routine allows several copies of the data to be printed out as required, and allows monthly records to be stored on tape, re-LOADed and used as needed. If this facility is to be used, the program should be started by entering GOTO 1, not RUN, as the latter will clear the stored variables.

Here follows a brief breakdown of the program:

Lines	Description
1-999	Set up the variables and operating instructions.
1000	Draws the axes for the graphs.
2000	Deals with maximum temperatures in degrees C.
3000	Deals with minimum temperatures in degrees C.
4000	Deals with percent humidity.
5000	Deals with pressure in millibars.
6000	Deals with hours of sun.
7000	Deals with inches of rain.
8000	Routine for printing out stored data.
9000	Error correction routine.

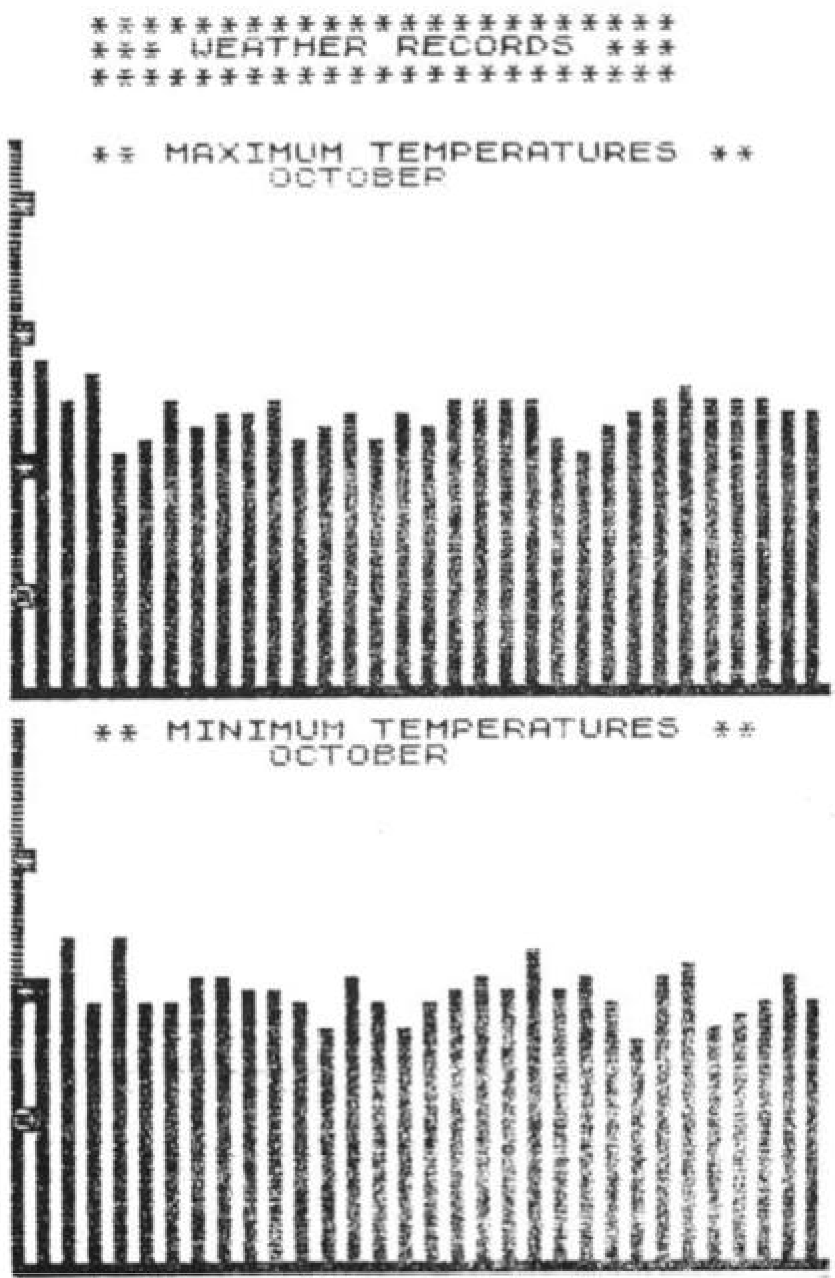
Singing in the rain

The range of allowed values for the different parameters can be altered by changing the calculations in the following lines:

2220 Maximum temperature.
3220 Minimum temperature.
4210 Humidity.
5210 Pressure.
6210 Sun.
7210 Rain.

The values set have been used for a year in London and found to be satisfactory, but in other areas changes may be needed, for example, to the rainfall; this range can be altered to 0 — 2 inches by changing the '40' in line 7210 to 20.

If the data for any particular day is missing entering '-100' will cause that data to be left blank in the final plot.



Examples of the output of this program — the minimum and maximum temperatures for one month.

```
  4 PRINT TAB 6;"WEATHER RECOR
D"
  6 LPRINT TAB 3;"*************
**********"
  8 LPRINT TAB 3;"*** WEATHER R
ECORDS ***"
 10 LPRINT TAB 3;"*************
**********"
 12 LPRINT
 14 LPRINT
 16 PRINT ,,"DO YOU WISH TO"
 17 PRINT ,,"(A) ENTER NEW DATA
"
 18 PRINT ,,"(B) TO PRINT OUT D
ATA STORED"
 19 PRINT ,,"ENTER A OR B"
 20 IF INKEY$="" THEN GOTO 20
 22 IF INKEY$="A" THEN GOTO 30
 24 IF INKEY$="B" THEN GOTO 800
0
 25 PRINT "ENTER A OR B"
 26 GOTO 20
 30 DIM T(31)
 40 DIM P(31)
 50 DIM S(31)
 60 DIM R(31)
 70 DIM M(31)
 80 DIM H(31)
 85 CLS
 90 PRINT
```

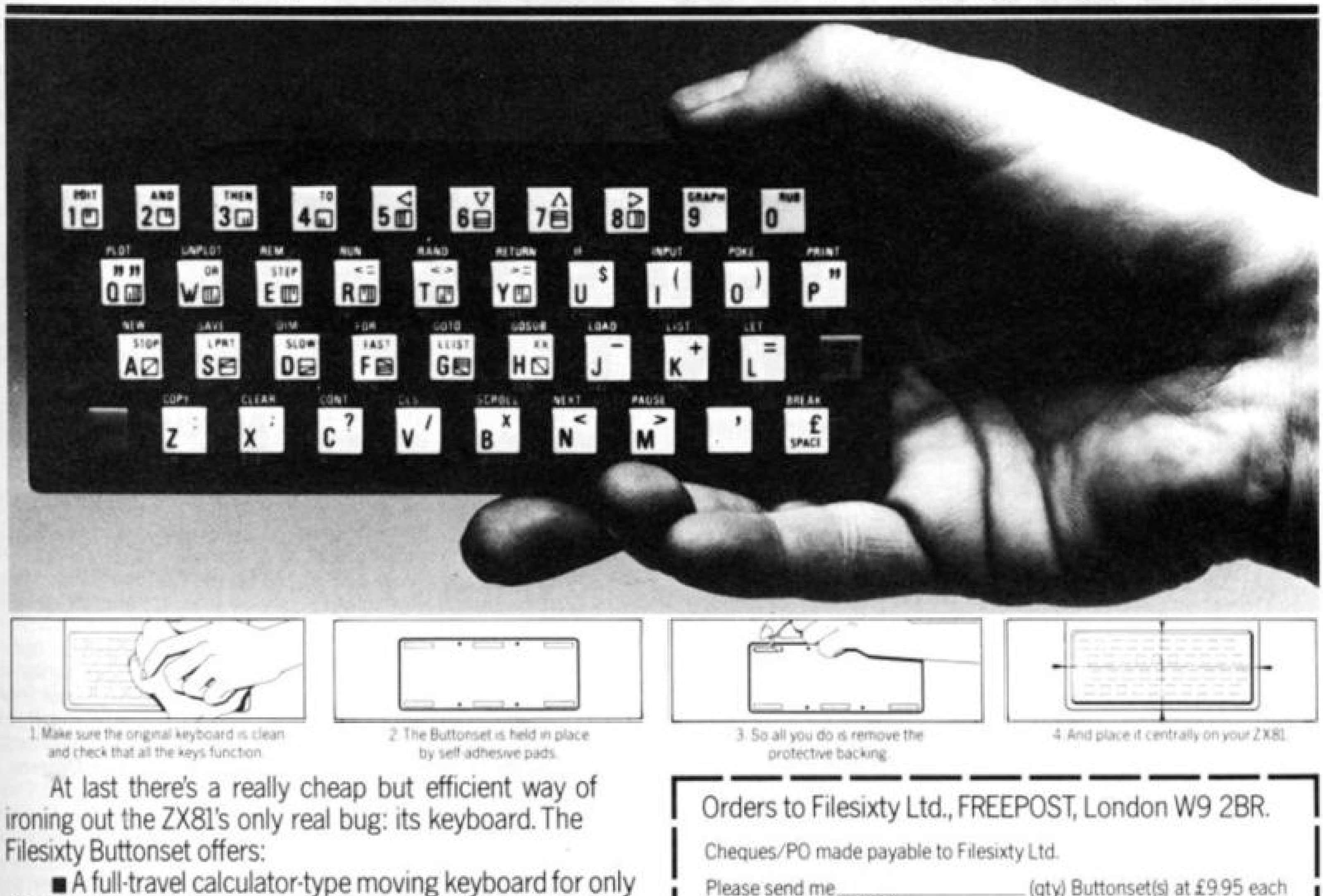


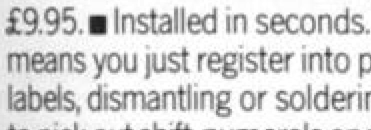
```
 100 PRINT "WHICH MONTH?   ";
 110 INPUT M$
 120 PRINT M$
 122 LET MONTH=LEN M$
 124 LET MONTH=32-MONTH
 126 LET MONTH=MONTH/2
 130 PRINT
 140 PRINT "HOW MANY DAYS?   ";
 150 INPUT DAYS
 160 PRINT DAYS
 170 PRINT
 175 CLS
 180 PRINT "WHAT TYPE OF DATA?
";
 190 PRINT
 200 PRINT TAB 5;"1. MAXIMUM TEM
PERATURE"
 210 PRINT TAB 5;"2. MINIMUM TEM
PERATURE"
 220 PRINT TAB 5;"3. HUMIDITY"
 230 PRINT TAB 5;"4. PRESSURE"
 240 PRINT TAB 5;"5. SUN"
 250 PRINT TAB 5;"6. RAIN"
 255 PRINT TAB 5;"7. END"
 260 PRINT "ENTER THE APPROPRIAT
E NUMBER"
 270 IF INKEY$="" THEN GOTO 270
 280 IF INKEY$="1" THEN GOTO 200
0
 290 IF INKEY$="2" THEN GOTO 300
0
 300 IF INKEY$="3" THEN GOTO 400
0
 310 IF INKEY$="4" THEN GOTO 500
0
 320 IF INKEY$="5" THEN GOTO 600
0
 330 IF INKEY$="6" THEN GOTO 700
0
 340 IF INKEY$="7" THEN GOTO 499
 499 STOP
 500 PRINT
 510 PRINT ,,"IF YOU ENTER A WRO
NG VALUE,"
 520 PRINT "ENTER -999 FOR THE N
EXT VALUE."
 530 RETURN
1000 REM TO PRINT AXES
1010 CLS
1020 FOR J=0 TO 21
1030 PRINT AT J,0;"■"
1040 NEXT J
1050 FOR J=0 TO 31
1060 PRINT AT 21,J;"■"
1070 NEXT J
1080 RETURN

2000 REM MAX. TEMPERATURES
2005 LPRINT
2006 LPRINT
2010 CLS
2015 LET LINE=1000
2020 PRINT AT 4,3;"MAXIMUM TEMPE
RATURES"
2030 PRINT AT 5,0;"I WILL PRINT
OUT THE DATES,",,"YOU ENTER THE M
AXIMUM",,"TEMPERATURE - "
2040 PRINT AT 9,0;"RANGE ALLOWED
 -6 TO 32"
2045 GOSUB 500
2050 FOR I=1 TO DAYS
2055 SCROLL
2060 PRINT I;
2070 INPUT T(I)
2075 IF T(I)=-999 THEN GOTO 9000
2080 PRINT TAB 6;T(I);TAB 11;"DE
GREES C"
2090 NEXT I
2100 GOSUB 1000
2110 PRINT AT 17,0;"■";AT 12,0;"
■";AT 7,0;"■";AT 2,0;"■"
2200 FOR I=2 TO (DAYS*2) STEP 2
2210 FOR J=T(I/2) TO -6 STEP -1
2220 PLOT I,J+6
2230 NEXT J
2240 NEXT I
2250 PRINT AT 0,3;"** MAXIMUM TE
MPERATURES **"
2260 PRINT AT 1,MONTH;M$
2280 COPY
2290 IF MARK=1 THEN GOTO 8020
2300 GOTO 175
3000 REM MINIMUM TEMPERATURE
3005 LPRINT
3006 LPRINT
3010 CLS
3015 LET LINE=2000
3020 PRINT AT 4,3;"MINIMUM TEMPE
RATURES"
3030 PRINT AT 5,0;"I WILL PRINT
OUT THE DATES,",,"YOU ENTER THE M
INIMUM",,"TEMPERATURE - "
3040 PRINT AT 9,0;"RANGE ALLOWED
 -10 TO 25"
3045 GOSUB 500
3050 FOR I=1 TO DAYS
3055 SCROLL
3060 PRINT I;
3070 INPUT M(I)
3075 IF M(I)=-999 THEN GOTO 9000
3080 PRINT TAB 6;M(I);TAB 11;"DE
GREES C"
3090 NEXT I
3100 GOSUB 1000
3110 PRINT AT 15,0;"■";AT 5,0;"■
";AT 10,0;"■"
3200 FOR I=2 TO (DAYS*2) STEP 2
3210 FOR J=M(I/2) TO -10 STEP -1
3220 PLOT I,J+12
3230 NEXT J
3240 NEXT I
3250 PRINT AT 0,3;"** MINIMUM TE
MPERATURES **"
```

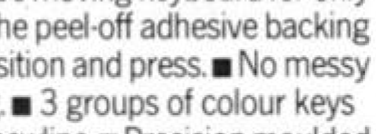
```
3270 PRINT AT 1,MONTH;M$
3280 COPY
3290 IF MARK=1 THEN GOTO 8050
3300 GOTO 175
4000 REM HUMIDITY
4005 LPRINT
4006 LPRINT
4010 CLS
4015 LET LINE=3000
4020 PRINT AT 4,10;"HUMIDITY"
4030 PRINT AT 5,0;"I WILL PRINT
OUT THE DATES,","YOU ENTER THE H
UMIDITY - "
4040 PRINT AT 9,0;"RANGE ALLOWED
 30 TO 100 PERCENT"
4045 GOSUB 500
4050 FOR I=1 TO DAYS
4055 SCROLL
4060 PRINT I;
4070 INPUT H(I)
4075 IF H(I)=-999 THEN GOTO 9000
4080 PRINT TAB 6;H(I);TAB 11;"PE
RCENT"
4090 NEXT I
4100 GOSUB 1000
4110 PRINT AT 0,0;"■";AT 6,0;"■"
;AT 12,0;"■";AT 17,0;"■"
4200 FOR I=2 TO (DAYS*2) STEP 2
4210 FOR J=H(I/2)/1.8-14 TO 1.5
STEP -1
4220 PLOT I,J
4230 NEXT J
4240 NEXT I
4300 PRINT AT 0,9;"** HUMIDITY *
*"
4310 PRINT AT 1,MONTH;M$
4320 COPY
4330 IF MARK=1 THEN GOTO 8080
4340 GOTO 175
5000 REM PRESSURE
5005 LPRINT
5006 LPRINT
5010 CLS
5015 LET LINE=4000
5020 PRINT AT 4,10;"PRESSURE"
5030 PRINT AT 5,0;"I WILL PRINT
OUT THE DATES,","YOU ENTER THE P
RESSURE - "
5040 PRINT AT 9,0;"RANGE 955 TO
1050 MILLIBARS"
5045 GOSUB 500
5050 FOR I=1 TO DAYS
5055 SCROLL
5060 PRINT I;
5070 INPUT P(I)
5075 IF P(I)=-999 THEN GOTO 9000
5080 PRINT TAB 6;P(I);TAB 12;" M
ILLIBARS"
5090 NEXT I
5100 GOSUB 1000
5110 PRINT AT 10,0;"■";AT 6,0;"■
";AT 14,0;"■"
5200 FOR I=2 TO (DAYS*2) STEP 2
5210 FOR J=P(I/2)/2.35-404 TO 1.
5 STEP -1
5220 PLOT I,J
5230 NEXT J
5240 NEXT I
5300 PRINT AT 0,9;"** PRESSURE *
*"
5310 PRINT AT 1,MONTH;M$
5320 COPY
5330 IF MARK=1 THEN GOTO 8110
5340 GOTO 175
6000 REM SUN
6005 LPRINT
6006 LPRINT
6010 CLS
6015 LET LINE=5000
6020 PRINT AT 4,12;"SUN"
6030 PRINT AT 5,0;"I WILL PRINT
OUT THE DATES,","YOU ENTER THE H
OURS OF SUN - "
6040 PRINT AT 9,0;"RANGE ALLOWED
 0 TO 16 HOURS"
6045 GOSUB 500
6050 FOR I=1 TO DAYS
6055 SCROLL
6060 PRINT I;
6070 INPUT S(I)
6075 IF S(I)=-999 THEN GOTO 9000
6080 PRINT TAB 6;S(I);TAB 11;"HO
URS"
6090 NEXT I
6100 GOSUB 1000
6110 PRINT AT 16,0;"■";AT 11,0;"
■"
6200 FOR I=2 TO (DAYS*2) STEP 2
6210 FOR J=S(I/2)*2.5+1 TO 1.5 S
TEP -1
6220 PLOT I,J
6230 NEXT J
6240 NEXT I
6300 PRINT AT 0,7;"** HOURS OF S
UN **"
6310 PRINT AT 1,MONTH;M$
6320 COPY
6330 IF MARK=1 THEN GOTO 8140
6340 GOTO 175
7000 REM RAIN
7005 LPRINT
7006 LPRINT
7010 CLS
7015 LET LINE=6000
7020 PRINT AT 4,12;"RAIN"
7030 PRINT AT 5,0;"I WILL PRINT
OUT THE DATES,","YOU ENTER THE I
NCHES OF RAIN - "
7040 PRINT AT 9,0;"RANGE ALLOWED
 0 TO 1 INCH"
7045 GOSUB 500
7050 FOR I=1 TO DAYS
7055 SCROLL
7060 PRINT I;
7070 INPUT R(I)
7075 IF R(I)=-999 THEN GOTO 9000
7080 PRINT TAB 6;R(I);TAB 11;"IN
CHES"
7090 NEXT I
7100 GOSUB 1000
7110 PRINT AT 0,0;"■";AT 20,0;"■
"
7200 FOR I=2 TO (DAYS*2) STEP 2
7210 FOR J=R(I/2)*40+1.3 TO 1.5
STEP -1
7220 PLOT I,J
7230 NEXT J
7240 NEXT I
7300 PRINT AT 0,6;"** INCHES OF
RAIN **"
7310 PRINT AT 1,MONTH;M$
7320 COPY
7330 IF MARK=1 THEN GOTO 8170
7340 GOTO 175
8000 REM TO PRINT OUT DATA ALREA
DY STORED
8005 LET MARK=1
8010 GOTO 2100
8020 LPRINT
8030 LPRINT
8040 GOTO 3100
8050 LPRINT
8060 LPRINT
8070 GOTO 4100
8080 LPRINT
8090 LPRINT
8100 GOTO 5100
8110 LPRINT
8120 LPRINT
8130 GOTO 6100
8140 LPRINT
8150 LPRINT
8160 GOTO 7100
8170 STOP
9000 REM ERROR CORRECTION
9010 IF I<=3 THEN LET I=1
9020 IF I>3 THEN LET I=I-3
9030 SCROLL
9040 PRINT "RE-ENTER FROM HERE -
"
9050 SCROLL
9060 GOTO 1055+LINE
```

NOW. A ZX81 PUSH-BUTTON KEYBOARD FOR UNDER £10.

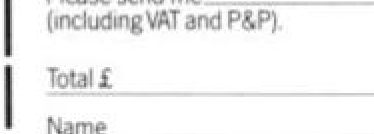




1. Make sure the original keyboard is clean and check that all the keys function.



2. The Buttonset is held in place by self-adhesive pads.



3. So all you do is remove the protective backing.



4. And place it centrally on your ZX81.

At last there's a really cheap but efficient way of ironing out the ZX81's only real bug: its keyboard. The Filesixty Buttonset offers:

■ A full-travel calculator-type moving keyboard for only £9.95. ■ Installed in seconds. The peel-off adhesive backing means you just register into position and press. ■ No messy labels, dismantling or soldering. ■ 3 groups of colour keys to pick out shift, numerals and newline. ■ Precision moulded in ABS to match your ZX81, with contrasting legends for maximum legibility.

Filesixty Ltd., 25 Chippenham Mews, London W9 2AN, England.
Tel: 01-289 3059. Telex: 268 048 EXTLDN G 4087.

Orders to Filesixty Ltd., FREEPOST, London W9 2BR.

Cheques/PO made payable to Filesixty Ltd.

Please send me ________ (qty) Buttonset(s) at £9.95 each (including VAT and P&P).

Total £ ________ BLOCK CAPITALS

Name ________

Address ________

____

ZXC 5/83

FILESIXTY

Twelve on trial

Our reviewer, James Walsh, takes a look at twelve new software packages for the ZX Spectrum. Do they come up to scratch?

In this review I shall be looking at a range of games from the fast action games of the arcade to the gripping tension of adventure games. They range from the graphically amazing to the graphically poor. It has also become painfully obvious just how misleading the packaging of a cassette can be.

Because each cassette is separate they shall be treated as such, in that a sub-review will be done of each, with a summary chart and conclusion at the end.

3D Tunnel — New Generation Software

This program comes from the author of the block-busting ZX81 games 3D Monster Maze and Defender, and this quality is reflected in 3D Tunnel. To set the scene: you are in a long winding tunnel infested by bats, rats and spiders, and even a London underground train (48K version only). As you move down the tunnel at one of the selected speeds, you can see a cross in the centre of the screen. It is possible to change your relationship to the tunnel via a joystick or various sets of keys on the keyboard. The idea of the game is to get to the other end of the tunnel by killing off or avoiding the strange creatures, as well as trying to keep within the tunnel.



The instructions are good though a little more about the different key configurations would have helped. The initial keyboard layout is very awkward as it seems to think that you are better at imagining a joystick than seeing your fingers on the keys.

The game itself is a breath of fresh air to me; it is not based around space, is original, and uses the graphics capabilities of the Spectrum to the full. The detail of the bats, rats and frogs which come at you is incredible — it is hard to believe that it is a Spectrum being used. If you have a 48K machine, you also have the task of having to avoid a London Underground train — the graphics are amazing.

There are three levels of play, from slow to fast, and the option of demonstration and practice runs at particular waves of attackers. The graphics are so good that I could have just sat and watched the demo for an hour. This is very professionally put together and a game that I would recommend to anyone.

Abacus Games Pack 2 — Abacus

There are four games on this cassette: Nine Lives, Bulls Tables, Candyman and Target. It is a mixed bag in more than one sense.

Nine Lives is a cat and mouse game, in which you must catch as many of the stationary mice as you can whilst evading the dog (who will eat you) but save the mice by carrying them to safety. Though the principal is simple, it is original and fun to play. The graphics are not quite crowd-stopping but are good; this is quite a good games pack type game.

The second program is called Bulls Tables in which you must answer ten sums of selectable type (addition, subtraction, multiplication and division), and difficulty, otherwise the bull will escape from his field and eat you. The graphics are very good with a little man running around the screen pick-

ing up and depositing numbers. The idea of having an angry bull trying to get out is, educationally, very good, as it keeps people (yes, people not just kids) interested. Unfortunately, this is typical example of it being more fun, graphically, to lose than win, as the bull runs around the screen and eats your numbers. Apart from this last point it is a good game.

Candyman is a game for two players playing separately, in which each player must run from the lower platform to the top without being hit by a piece of psychedelic scaffolding; if successful then you have to get your next man up. Again this game is original and often quite difficult (it is probably the best on the tape) though the graphics are a little disappointing. It is definitely fun and more addictive than many 'games pack' games.

Finally, there is Target, which deserves last place, as it is very predictably a game of shooting the ducks, etc, as they go across the top of the screen. The graphics are quite good but the rifle is so far away that absolutely no skill is possible.

Overall, this games pack is good but not the best. It is fun but it is not all particularly well-written. Bulls Tables is the odd one out in as much as it should be on an educational pack. However, quite a fun set of programs.

Horace Goes Skiing - Sinclair Research/ Psion

It looks rather as if Uncle Clive is taking Horace as a sort of semi-mascot, as this is the second in the series of programs in which he has starred.

In the first part, Horace must cross the road and buy his skies without getting run over by the on surge of traffic. He then pays $10 for a pair of skiis (yes dollars, what have they got against the pound sterling?). Horace then re-crosses the road, and if he is still alive, a skiing game commences.

The graphics all the way through are excellent. Though both these have been done before separately, never before have the graphics been so good on one tape. Because they have managed to bring extra life and enjoyment into these games, they must be the best representations of either on a low-budget computer. Using the two games in conjunction produces one of the most addictive packages I have yet encountered.

Horace Goes Skiing definitely helps set the new higher standard for Sinclair distributed software.

Tobor — Elfin Software



You are in a maze protected by robots. The aim is to destroy these either by shooting them or causing them to collide and hence destroy themselves. They continuously change colour and depending on this, your score is calculated. If a robot goes green then it is indestructable until it changes again, so it is a good idea to keep out of its way. When one lot of robots have been destroyed then a new maze and robots are created. If you are killed, a small tombstone is built, and another of your three lives is lost. Tobor can also be used as a two player game, in which the players take it in turn to try their skill against the robots.

Though the principal is pretty straightforward, the graphics are good and the way in which the game has been put together is good and very professional indeed. It is definitely a program that I would recommend.

As far as I know, Elfin Software have not been in existence for long, though they have a good range of Spectrum programs, including: 3D Starwars, Checkman, 3D Game Hunt, Pilot and Pack Man. I very much look forward to receiving these titles as, if they come up to the high standard of Tobor, then they too should be very good value for money. My only grievance would be that the instructions were rather limited.

Rescue — Computer Rentals Limited

Rescue by D. Barker is another adventure type game. It is a little different, in that a limited amount of graphics are used. The object is to get from the inner ring to the outer ring and then to the castle where you make a daring rescue and then escape back to base.

Graphics are used when you look at a map of where you are. Also if you find the radio then you are able to find out where the enemy are. However, if you arrive at a place occupied by the enemy and you do not have a gun or a uniform, then you are automatically dead, which seems a shame really. The game is real-time, hence the enemy will move even if you don't.

The instructions are pretty long so they have to be put on a separate program. This is rather annoying because there is no instruction form or leaflet, so if you forget something then there is no easy way of looking it up whilst playing the game. Quite a good game, if you like that sort of thing. By the way, the packaging is nice.

ZX Trek — Impact Software

Peter Lovett has written a new advanced version of the well-known game, Star Trek, which is in many ways the thinking person's Space Invaders, though I do not mean to denigrate all those millions of Space Invader fans. It is true that th idea of the 'Star Trek' game has become rather uninspired lately because of the rather poor quality of many of the ZX81 versions. Thankfully, this trend has come to an abrupt end with the announcement of this new Spectrum version. ZX Trek is a real-time game so no dozing off at the console. I am pleased to say that the documentation is long and thorough, and it needs to be!

Your task is to free the galaxy of the enemy. Sounds easy? Well, it sure ain't. You are situated in endless space. Your own galaxy occupies 100 quadrants within this space. A whole galaxy may sound cushy, but not with up to 70 Klingons, 2,000 stars and just a handful of bases. You have full control of your craft with its advanced computers and its repair crew. If you are attacked then the amount of damage sustained is proportional to the strength of the shields. The on-board computer gives you a full report on the effectiveness of all your equipment. Unfortunately, if the ship's own computer gets damaged then this information may not be totally accurate. Your own crew undertake repairs on any part of the ship, though if you can get to a starbase then the repair can be carried out around four



times faster. You have full use of torpedos and phasers, and if by any chance you run out of Dilithium Crystals you can land on a star and search for replacements.

The display makes good use of colour for representation of status which brightens up the display tremendously. The option of additional sound effects is also provided so that conditions of danger are more obvious.

This is a game which one could carry on playing for hours. I had limited success on level three and dread to think what might happen on the highest of the ten levels, level 9!!! If you are interested in a game in which chance is not the only factor and which you will be able to play again and again without coming up against the same circumstances, then ZX Trek should suit you. I do feel that ZX Trek will find its way into many a home, as it is by far the best version I have seen so far.

By the way. I liked the introductory screen, nothing to do with 'Star Trek', but definitely stunning.

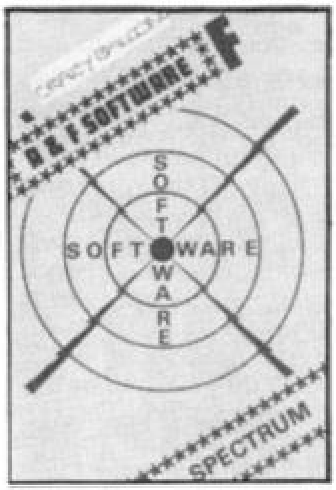
Cruising On Broadway — Solarsoft



The idea of Cruising is to get round a course, which is a single pixel line, without being caught by the computer's opposition car. The inital course is a double figure of eight, whilst the subsequent courses, which you get onto once the previous one has been completed, are far more random (there are four in all).

Initially, the game is disappointing with only coloured squares as cars and a single line as the track. Also, you only have one life per game; therefore, the game can be rather short. If you get past the lack of initial impact you should find yourself enjoying this simple game. This game is good in its addictive quality, but lacks the substance really to be sold on its own. If there were two games of equivalent standard to this on the tape, then it would be better value. Apart from this, you can not take away the fact that it is quite definitely fun to play.

Crazy Ballons — A & F Software



The idea of Crazy Balloons is to negotiate a hot air balloon around the course shown on the screen. If you manage this then the course is made harder by the addition of moving blocks. If you get further still then the blocks appear randomly over the screen trying to cause you to crash. At certain points along the course there are short cuts for which you get extra points, but they are often near impossible to negotiate. There is a set number of points for completing a course and a time bonus depending on speed.

Though the idea of the game may seem simple, it is far from easy to play. It is also highly addictive. Unfortunately it does not quite have the substance to stand up for itself on a cassette.

A good original game which is fun but which does not, on its own, stand up to some of the better software looked at in this review so far.

Do Not Pass Go — Workforce

Do Not Pass Go is a computer implementation of the popular board game, Monopoly. All the facilities (except for knocking off the opposition's houses) are available for between two and six players. The main disappointment for me was that the computer itself does not play. Surely this would not been too difficult!

As it is, Do Not Pass Go is really only an alternative to the usual card and plastic pieces. The screen itself is rather cramped — too much has been put into a small space; perhaps they should have used more of the screen. Also, no actual indication on the board is given of where you are — it is necessary to fathom it out from a printed number which corresponds to a square. Though, in some ways this program has been well done, the programmer has made it rather limited and has made some crucial mistakes in presentation. If, via the computer, it was possible to play Monoploy on your own then this could be a winner.

I am also disappointed in this program when I look at the various other pieces of software they have on the market, for example, Base Invaders and High Noon, which are probably the best versions of their respective games for the Spectrum. They both use High-Res and colour to its full, and are fast.

Personally, having played Base Invaders and particularly High Noon quite extensively, I can happily recommend them as excellent examples of professionally produced games. But for Do Not Pass Go, I can only say buy it if you have not got the board game, or you really like the idea of using the computer and are struck by the idea of playing Monopoly.

Cosmos — Abbex



Abbex came into the market back in October of last year with two Spectrum games programs. We are yet (April) to see anything new from them which seems to suggest that the original games are of exceptional quality. Of the two, Spookyman and Cosmos, the only one that really caught my eye was Cosmos (surely the world has seen enough 'Pacman' copies by now!).

The scenario of the game is as follows: you are in charge of a fleet of ships which must be protected from successive waves of aliens and from meteorites which just happen to come along. Each time either a metorite or an alien hits a ship in the fleet it is destroyed. If you manage to destroy the total population of aliens within a quadrant of the fleet then another wave will appear in greater numbers, and the fleet will be regenerated.

Graphically the game is very good and highly addictive. The only niggles that I have are that I would have liked different space ships for each wave and secondly, your movements around the quadrant could be smoother. Even taking these into account, it is a very well-assembled game, which is just different enough to interest someone who has already seen many other space games and become addicted. Recommended.

P.S.E. Games Tape 3 — P.S.E.

On this tape are three games written by Steve McCarthy, some of which are up-grades of ZX81 games, whilst others are new. The game are as follows: 3D Noughts and Crosses, Towers of Brahm, Wipe-out and the flag ship of the pack, Astro-Wars.

3D OXO is an obvious game in which there is a four by four by four matrix in which it is necessary to get four in a row in any direction to coin. The only problem being that the layers are separated for display which takes away some of the effect. The computer itself is a pretty formidable opponent. Quite a well-presented thinking person's game.

Secondly, comes Towers of Brahm, the age old problem of getting five discs of ascending size from pin one to pin three. Though the principle is simple, the game is fast and colourful. This is an enjoyable game to play which is more addictive than you might at first expect. User-defined graphics could have been better used; this should not have slowed things down much if at all.

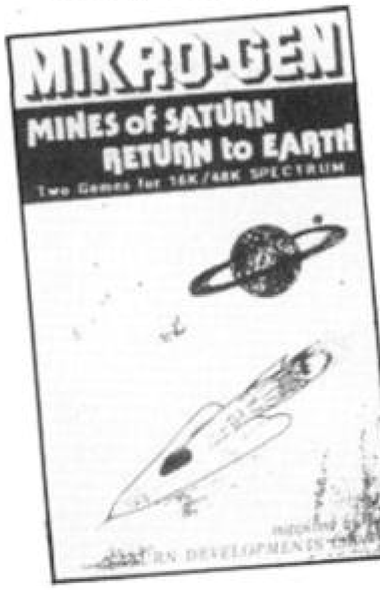
Wipe-Out is the final game on side A. The title seems to suggest that it may in some way be related to the hoards of space invader 'zap-zap' games, but far from it. This is a two player game, again putting an emphasis on thought. The com-

puter draws a matrix board with its centre and corners made up of squares Xs and Os at the edges, and a grid reference system around it. The first symbol of each game to be 'wiped out' must be a square. Each successive grid reference thereafter specified must be immediately adjacent to the one previously 'wiped out' by either player. The object is to wipe out as many of either Xs or Os depending on which side you are on. This game is far from simple, though it would be greatly improved if it was possible to play the computer. Graphics are used very sparingly on this, but it is fun if you are into thinking games.

Finally, on side B we come to Astro-Wars which is a complete departure from side A in that it is a fast action, arcade game. It is an all machine code game in three stages based on the films 'Star Wars' and 'Empire Strikes Back'. The overall object being to score as many points as possible, but other objectives and hazards are generated on the way. In stage one you are looking out of your ship and into space and enemy aircraft are moving onto and around the screen. It is possible to destroy them via missiles and a sight which moves about the screen. A discrepancy which is noticeable pretty quickly is the fact that the rays of the missiles seem to go from the target to you, the ship, rather than *vice versa*. Whether or not you survive this, you are thrust into a long 3D tunnel in which missiles and other objects are projected at you in a 2D plane at the far end. Your task is to survive for six minutes until the Death Star comes into sight, at which time you must try to destroy it. Again, even if you fail on this stage you move onto the final stage, in which Imperial Walkers are coming at you in 3D and enemy aircraft are in the distance. They are both firing at you, which makes life very difficult indeed!

Overall, this game is easily the best on this package. The games are good but not excellent separately, though Astro-Wars is quite well-written. The documentation is very good and as a pack of four programs it is very good value indeed. I would recommend you to look out for any new stuff written by Steve McCarthy and Precision Engineering, as the quality is definitely improving at a fantastic rate. This tape can be recommended as good value for money.

Mines Of Saturn / Return To Earth — MikroGen



There has always been a heavy presence of adventure games on the ZX81 and Spectrum market, though recently the standard of the best has improved fantastically with the launch of The Hobbit. For this reason, it is very important that if you want to market an adventure, that it must be very good to stand any chance of success. It is also important to use some if not all machine code and a 48K machine to make the possible permutations large enough to make the game interesting.

The plots in both Mines of Saturn and Return to Earth are very predictable. Neither of them use any graphics at all in the games themselves which seems a pity considering that you are dealing with a computer with excellent potential. They are both for 16K machines though only take up 7K out of the 9K available. The vocabulary is very small, as is the number of possibilities. It is very easy to get killed as you have no way at all to fight back. The sentence interpretation routine is very poor and slow — another thing that you should not get on the Spectrum. If you lose then it is necessary to type GOTO 1 to re-start, surely a 'Do you want another game?' routine would not have strained the programmer too much.

Basically, the quality of this game does not even come up to that of the better ZX81 adventures. Definitely not one that I could recommend.

Summary

By looking at the summary table you will be able to see how these programs compare, and how we are still in a situation where there is little excellence. I must commend both New Generation as well as Sinclair Research for working to set new high standards of software.

Quality is definitely going up at an incredible rate, but some companies are being left behind.

SUMMARY CHART

Product Name	Price (£)	Memory required	Documentation	Addictive quality	Use of graphics	Programming achievement	Value
Crazy Balloons	5.75	16K	3	3½	3	3	3
Cosmos	4.95	16K	3	4	4	4	4
Tobor	7.95	48K	3	4	3½	3½	3½
Horace Goes Skiing	5.95	16K	4	4½	5	4½	4½
Rescue	5.95	48K	3	3	3	3	3
Cruising on Broadway	4.95	16K	3	4	3	2½	3½
ZX Trek	6.50	48K	5	4½	3½	4	4½
Mines of Saturn /Return to Earth	5.95	16K	2½	2	0	2½	2½
PSE Games Tape 3	4.95	16K	4	3½	4	3	3½
Do Not Pass Go	4.95	48K	5	4	3½	3	3½
Abacus Games Pack 2	4.95	16K	3½	4	4	2-4	3½
3D Tunnel	5.95	16/48K	4	5	5	5	5

Product Name	Supplier
Crazy Balloons	A&F Software, 830 Hyde Road, Manchester, M18 7JD.
Cosmos	Abbex Electronics Ltd, 20 Ashley Court, Great Northway, London NW5.
Tobor	Elfin Software, Hudson House, Battery Road, Great Yarmouth, NR30 3NN.
Horace Goes Skiing	Sinclair Research Ltd, Freepost, Camberley, Surrey.
Rescue	Computer Rentals Ltd, 140 Whitechapel Road, London E1.
Cruising on Broadway	Sunshine Books, 19 Whitcomb Street, London WC2 7HF.
ZX Trek	Impact Software, 70 Redford Avenue, Edinburough, EH13 OBW.
Mines of Saturn/ Return To Earth	MikroGen, 24 Agar Crescent, Bracknell, Berkshire.
P.S.E. Games Tape '3'	P.S.E., 20 Shelton Avenue, Newark, Notts.
Do Not Pass Go	Work Force, 140 Wilsden Avenue, Luton, Beds.
Abacus Games Pack	Abacus, 186 St. Helens Avenue, Swansea, W. Glamorgan.
3D Tunnel	New Generation Software, FREEPOST (BS3433), Oldland Common, Bristol, BS15 6BR.

Battle of Britain

A strategic Wargame for the 48K Spectrum

One day in summer 1940. Reports are coming in of enemy bomber squadrons crossing the English coast: target unknown.

How are you to deploy the nine fighter squadrons under your command to intercept the incoming threat!

British and enemy squadron movements are plotted on a superb high resolution screen map of south east England with communications signals presented both visually and in morse code.

Features variable difficulty levels, full control of squadron movements, randomly selected targets and bomber routes for each game, intelligence reports, refuelling etc.

A tense game of strategy for one player.



48K Strategic Wargame

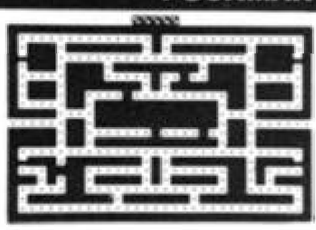
only £5.95 (inc. p&p)

MS

73 The Broadway, Grantchester, Cambridge CB3 9NQ

SF GOBBLERS

PUCKMAN FOR 16K ZX81



BEAT THAT HIGH SCORE! GOBBLE THOSE DOTS BEFORE THOSE MEANIES GOBBLE YOU! YOUR ONLY AIDES ARE FOUR "POWER PILLS" WHICH MAKE THE MEANIES EDIBLE. BUT NOT FOR LONG!

- MACHINE CODED FOR FAST ACTION
- EXTRA "GOBBLER" FOR 10,000 POINTS
- ON SCREEN SCORING
- HIGH SCORE WITH "ENTER NAME" FACILITY
- UP TO 4 PLAYERS

AN ANNOYINGLY FRUSTRATING GAME! FOR ONLY £5.95

SF ASTEROIDS

FOR 16K ZX81

STAY ALIVE AS LONG AS POSSIBLE IN OPEN SPACE FILLED WITH FLYING ROCKS. SCORE BY SHOOTING THEM – WHICH ALSO CAUSES THEM TO BREAK INTO LOTS OF LITTLE BITS AND MAKES LIFE EVEN WORSE!

- MACHINE CODED FOR FAST ACTION
- ON SCREEN SCORING
- HIGH SCORE WITH 'ENTER NAME' FACILITY
- UP TO 4 PLAYERS
- EXTRA SHIP FOR 1,000 PTS (NOT AS EASY AS IT SOUNDS!)
- SHIP MOVES JUST LIKE ARCADE VERSION
- ROTATE LEFT/ROTATE RIGHT/THRUST
- FIRES IN ALL 8 DIRECTIONS
- INCREASING NUMBER OF ASTEROIDS
- THREE ASTEROID SIZES
- 'NASTY' ALIEN SPACE-SHIP (FIRES BACK!)

THIS GAME IS JUST AS BAD! – AND ONLY £5.95

AN OFFER FOR REAL MASOCHISTS – BOTH TAPES FOR £9.95

MAIL ORDER ONLY–PLEASE MAKE CHEQUE/PO PAYABLE TO

THE SOFTWARE FARM DEPT A

CRAIGO FARM, BOTANY BAY, TINTERN, GWENT

New from ...

MIKRO-GEN

SPECTRUM Software

COSMIC RAIDERS (16K) £5.95



You — the only remaining pilot of the "Earth Defence Fleet" — have the daunting task of defending the Earth from attack by alien beings that have already set up a base of their own on earth.

* Radar Scanner display in hi-res *
* Full colour and sound effects *
* From the author of Spectrum 'Scramble' *
* 100% machine code *

SPACE ZOMBIES (16K) £5.95

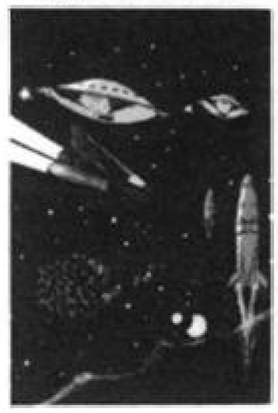


Shooting through space you are ambushed by a fleet of Space Zombies flying at you in formation, swooping and diving at you. Destroy them, if you can! But in destroying them, you attract more to the area, and the game gets progressively harder.

* Full screen hi-res graphics *
* 1 or 2 players * Joystick option *
* Full sound and colour *
* Three playing speeds *

MINES OF SATURN and RETURN TO EARTH £5.95

(16K) Text Adventure



Mines of Saturn While piloting a routine orbit of Saturn, you are caught up in a radiation storm which forces you into the giant planet's rings. Your energy drained, you make a forced landing on the planet's surface. Luckily you crashed near an abandoned mining base and you set off in search of some di-lithium crystals to refuel your stranded space ship. Can you do it?

Return to Earth Having escaped from your previous dilemmas, you reach Earth Station 1, but fail to make radio contact. You effect a safe if harrowing manual docking. On entry you find it deserted, and the control room destroyed. You must explore the station and find some way to alert Earth of your predicament, but beware, many of the rooms are identical, there is extensive damage, and signs of alien intruders.

MAD MARTHA £6.95

Graphic Adventure (48K)



Poor little Henry is the hen-pecked hero of this domestic tale. One night he can take no more. He steals his wages from his wife's purse, sneaks out of the house being careful not to wake the baby or trip over the cat, and heads for the bright lights to have some fun. Trouble is he runs out of money. His only way to raise some cash is to gamble his few remaining pounds on the spinning wheel of the roulette table. Just as Henry is getting into his evening his wife — Mad Martha — has noticed his absence. Realising Henry has absconded with the housekeeping she sets out after him with an axe. Guess what part you play in this happy tale? That's right, you're Henry. Watch out for that axe!

Access BARCLAYCARD VISA

Access or Visa Card holders please telephone (0344) 27317 (9am–6pm)

Please add 40p per order for post & packing

Name ______ Address ______

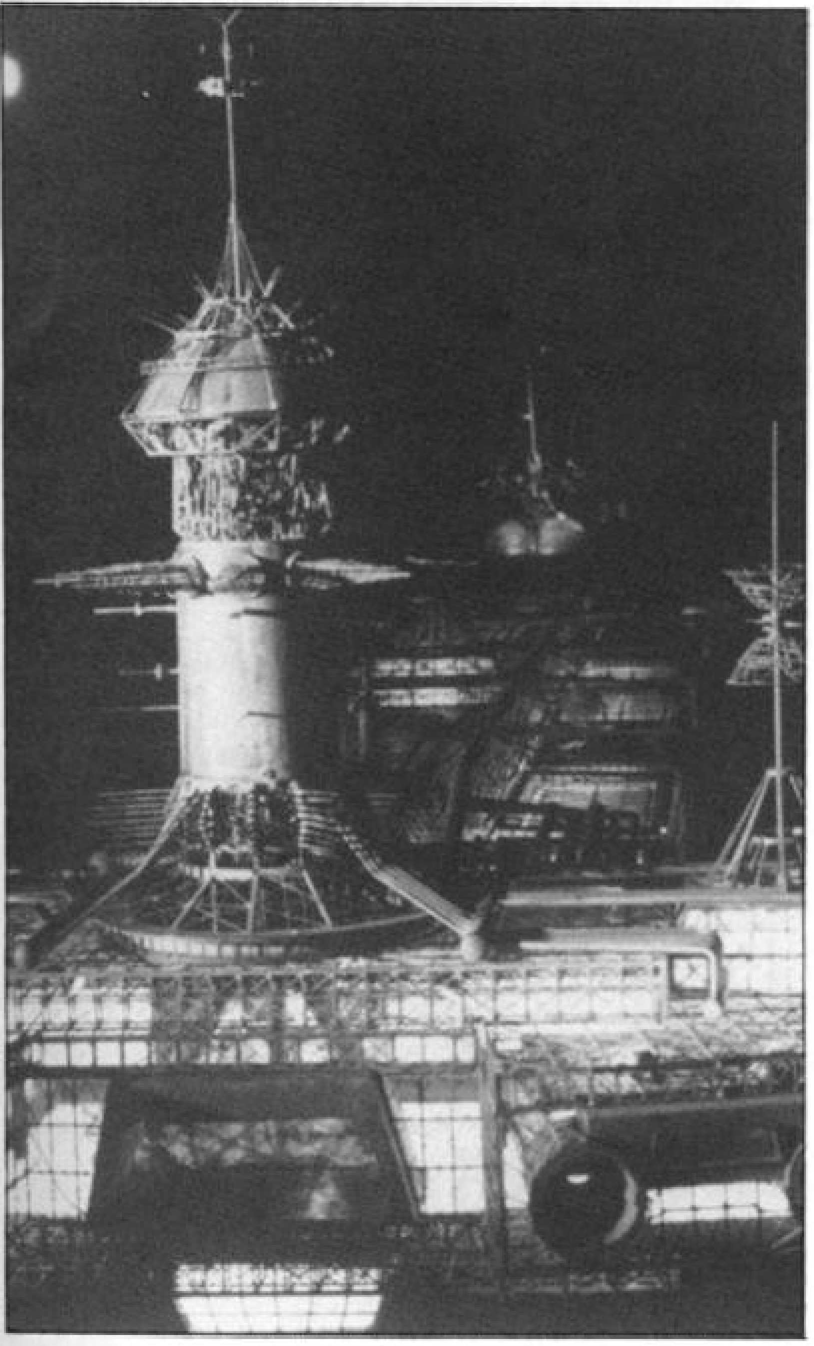
☐ Cosmic Raiders ☐ Space Zombies

☐ Mines of Saturn/Return to Earth ☐ Mad Martha

MIKROGEN, Dept A3, 24 Agar Crescent, Bracknell, Berkshire, RG12 2BK

Meteors II

Join the space race with Neil Streeter and his ZX81.


MCMLXXIX Walt Disney Productions.

I first got my ZX81 about six weeks after the launch of the machine way back in April 1981. I set down to writing 1K programs for it and, after a couple of weeks, I had a selection which I sent off for publication. Eight of these were accepted for publication, one of which (called Meteors) provided the inspiration for this program.

The original Meteors program used the ZX81's SCROLL command to give a full moving display in 1K. The idea of the game was to move your space ship left and right in an effort to avoid the meteors and alien space craft coming at you. The trouble was that the background was white, which didn't give a very good impression of space! Also, the SCROLL function went the wrong way, which meant that it felt like you were moving towards yourself.

Two in one

Also, around the time, there were a number of 'racer' programs which used the same idea to represent a car travelling along a winding track. What I always wanted to do was to combine these two games, and with 16K that's exactly what I set out to do. After trying a BASIC version and finding you could make a cup of tea between moves, I decided to write the bulk of the program in machine code.

The game, entitled Meteors II after its elder brother, is very easy to play and is extremely addictive. The idea of the game is to pilot your space craft through various types of obstacles and your score, which appears on screen, depends on how long you manage to avoid disaster.

There are seven stages to be negotiated after which the whole game repeats — so, theoretically, you could be up in space forever!

The seven stages of the game are:

Star Field: This is the first stage, and as you would imagine easiest of the lot. All you have to do is to avoid the stars that head down at you from the top of the screen.

Rock Field: This scenario contains block shaped meteors that bear down at you from the top of the screen.

Asteroid Field: Similar to the last scenario, but this time the rocks are twice the size.

Meteor Field: Again similar to the last two 'fields' but now the rocks are three times the size and obviously harder to avoid.

Aliens: You now have a change of scenery as you find a mass of alien space craft heading towards you. This stage of the game is not as difficult as the stage before, but will provide you with a quick breather in preparation for the next two stages.

Super Birds: These are a weird alien life form and extremely large at that! These creatures are four times the size of the original rocks in the second 'field'.

The Tunnel: You now find yourself in a tunnel (similar to the 'racer' games I was talking about earlier). The tunnel gets narrower and narrower the further along you get. If you manage to make it to the other end, then you find yourself back at stage one of the game.

For each time you go round the seven stages, your score will be 7,200. I have so far managed to get through the game just over three times, a score of 21,750 — beat that if you can!

What's the code?

The program is fairly short and if you are careful with the machine code shouldn't present any difficulties. The purpose of the machine code subroutine is as follows:

— It scrolls the screen down one line.
— It moves the ship left or right one place if keys '8' or '5' have been pressed.
— It increases the score by ten.
— It checks whether anything has been hit.
— If anything has been hit then it returns a value of one; if not, then a value of zero is returned. Thus, all that is left for the BASIC program is to print the aliens at the top of the screen and jump to a crash and explosion routine if the USR routine is equal to one. Also included in the program are user prompts and instructions, to make the program extremely user friendly.

The machine code subroutine could be used in any BASIC program of a similar format.

Certain bytes may be changed as desired so that it can be used in other ways. Certain conditions must, however, be observed. These are:

— The score must appear as it does in this program, at the top left of the screen in inverse characters.
— The background code must appear in bytes 16621, 16649, 16661, 16671, 16677 and 16682.
— The code of the object being moved must be that that appears in byte 16666 of the machine code.
— The keys for left and right are tested for in bytes 16605 (left) and 16633 (right) and the codes for the keys being used must appear in these locations.

The machine code routine is called by:

LET A = USR 16518

Initially to set the position of the ship in bytes 16514 and 16515 and then called on all subsequent occasions by:

LET A = USR 16528

This will return a value of one or zero in A depending on whether anything has been hit or not.

Line by line

The graphics characters in the program are as follows:

— All the grey characters in the program are the graphics character on the 'H' key.

– Line 220 has 'graphics T graphics Y' in it.
– Line 225 has 'graphics G graphics 6 inverse V graphics 6 graphics G' in it. These are the graphics for the Super Birds.

The machine code itself can be loaded using the machine code loader program given, which on RUNning will prompt with the address and you will have to enter the decimal codes for those addresses. These codes are given in the third and sixth columns of the machine code listing.

The main parts of the program are shown in Table 1. Note the inverse I in line 3000 should be a normal I before SAVEing. To SAVE the program after entering, start the tape recorder and type 'GOTO 3000' as a direct command and press Newline. The program will autoRUN after SAVEing and on subsequent LOADings. A disassembly of the machine code follows in the hope that the subroutines within it will prove helpful to others.

Line 1	Machine code subroutine.
Line 4	Jumps to instructions and initialisation.
Lines 70-110	Main display routine.
Line 120	GOes TO the crash routine.
Lines 200-270	Changes character being scrolled routine.
Lines 300-450	Crash and replay routines.
Lines 700-750	Tunnel routine.
Lines 1000-1180	Initialisation.
Lines 3000-3020	SAVE for autoRUN.

Table 1.

```
   1 REM >=? CLEAR ?E£RND) ";b A
 DE£RND)?"; FAST "57( CLEAR FAST
 SGN LPRINT :> 5$4 PRINT E£RND)£
 ;Q F/ PRINT ?E RND)5 ;6 RNDLN
 "???=Y C LN TAB RUN RNDE RNDF6
 RND?(INKEY$? RETURN STAB (INKEY
 $E RND?6 RNDE RND RND TAN
 E RND " TAN UUUUUUUUUUUUUUUUU
 UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
 UUUU
   4 GOTO 1000
  70 PRINT AT 2,INT (RND*(33-LEN
 A$));A$
  80 LET A=USR 16528
  90 LET S=S+1
 100 IF S/100=INT (S/100) THEN G
OTO 200
 110 IF NOT A THEN GOTO 70
 120 GOTO 300
 200 IF A$(1)="[illegible]" THEN LET A$=A$
+"[illegible]"
 210 IF A$(1)<>"[illegible]" THEN LET A$="
[illegible]"
 220 IF S/400=INT (S/400) THEN L
ET A$="[illegible]"
 225 IF S/500=INT (S/500) THEN L
ET A$="[illegible]"
 250 IF S>=500 THEN PRINT AT 1,0
;"[illegible]
[illegible]"
 260 IF S>=600 THEN GOTO 500
 270 GOTO 70
 300 LET P=PEEK 16514+256*PEEK 1
6515
 305 FOR X=1 TO 3
 310 POKE P,0
 320 POKE P,128
 330 POKE P,CODE "*"
 335 FOR Y=1 TO 2
 336 NEXT Y
 340 POKE P,CODE "[illegible]"
 350 POKE P,0
 355 POKE P,128
 360 NEXT X
 370 LET S=(S+1)*10
 375 LET S=S+S1*7200
 380 IF S>HI THEN LET HI=S
 385 LET I$=STR$ HI
 390 LET H$=H$( TO 6-LEN I$)
 400 FOR X=1 TO LEN I$
 410 LET H$=H$+CHR$ (CODE I$(X)+
128)
 420 NEXT X
 430 PRINT AT 20,1;"PRESS ANY KE
Y TO PLAY AGAIN."
 440 IF INKEY$<>"" THEN GOTO 440
 450 IF INKEY$="" THEN GOTO 450
 460 GOTO 1100
 500 LET A$="[illegible]"
 510 LET T=12
 520 LET T=T+INT (RND*3-1)
 540 IF T<0 THEN LET T=0
 550 IF T>31-LEN A$ THEN LET T=3
1-LEN A$
 560 PRINT AT 2,T;A$
 570 LET A=USR 16528
 580 LET S=S+1
 590 IF S/15=INT (S/15) THEN LET
 A$=A$(2 TO )
 600 IF A$="[illegible]" THEN GOTO 700
 610 IF NOT A THEN GOTO 530
 700 PRINT AT 1,0;"[illegible]
[illegible]"
 710 PRINT AT 2,T-1;"[illegible]"
 720 LET S1=S1+1
 730 LET S=0
 740 LET A$="[illegible]"
 750 GOTO 70
1000 LET H$="000000"
1010 LET HI=0
1020 GOTO 2000
1100 PRINT AT 0,0;"SCORE 000000
[illegible]";I$;"[illegible]"
1110 FOR X=1 TO 20
1120 PRINT "[illegible]
[illegible]"
1130 NEXT X
1140 LET S=0
1150 LET S1=0
1160 LET A=USR 16516
1170 LET A$="[illegible]"
1180 GOTO 70
2000 PRINT TAB 10;"METEORS II."
2010 PRINT TAB 10;"[illegible]"
2020 PRINT "YOU ARE IN CONTROL O
F A TRADING SPACE SHIP WHICH SPE
NDS MOST OF"
2030 PRINT "IT""S TIME IN THE AS
TEROID BELTS."
2040 PRINT "YOU MUST NOT HIT ANY
 OBSTACLES AS YOU NEGOTIATE SPA
CE USING KEY ""5"" FOR LEFT A
ND KEY ""8"""
2050 PRINT "FOR RIGHT."
2060 PRINT " IF YOU CAN GET TO T
HE END OF THE INTER SPACE TUNN
EL THEN YOU WILL START AGAIN ON
THE OTHER SIDE."
2070 PRINT ,,TAB 11;"GOOD LUCK"
2080 PRINT ,,"PRESS ANY KEY TO S
TART PLAY...."
2090 PRINT ,,"[illegible]"
2100 IF INKEY$<>"" THEN GOTO 210
0
2110 IF INKEY$="" THEN GOTO 2110
2120 CLS
2130 GOTO 1100
3000 SAVE "METEORS I[illegible]"
3010 CLS
3020 RUN
```

The BASIC part of the listing.

```
   1 REM XXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXX
   5 SCROLL
  10 FOR X=16514 TO 16686
  20 PRINT X;" ";
  30 INPUT A
  40 POKE X,A
  50 PRINT A
  60 SCROLL
  70 NEXT X
```

The machine code loader program.

```
16514 00 0     16515 00 0
16516 00 0     16517 00 0
16518 2A 42    16519 0C 12
16520 40 64    16521 11 17
16522 00 0     16523 02 2
16524 19 25    16525 22 34
16526 82 130   16527 40 64
16528 2A 42    16529 0C 12
16530 40 64    16531 11 17
16532 72 114   16533 02 2
16534 19 25    16535 E5 229
16536 06 6     16537 21 33
16538 23 35    16539 10 16
16540 FD 253   16541 E5 229
16542 D1 209   16543 E1 225
16544 0E 14    16545 12 18
16546 06 6     16547 21 33
16548 7E 126   16549 12 18
16550 1B 27    16551 2B 43
16552 10 16    16553 FA 250
16554 0D 13    16555 20 32
16556 F5 245   16557 2A 42
16558 0C 12    16559 40 64
16560 11 17    16561 0C 12
16562 00 0     16563 19 25
16564 7E 126   16565 3C 60
16566 FE 254   16567 A6 166
16568 20 32    16569 05 5
16570 36 54    16571 9C 156
16572 2B 43    16573 18 24
16574 F5 245   16575 77 119
16576 2A 42    16577 82 130
16578 40 64    16579 11 17
16580 21 33    16581 00 0
16582 19 25    16583 22 34
16584 84 132   16585 40 64
16586 CD 205   16587 BB 187
16588 02 2     16589 44 68
16590 4D 77    16591 51 81
16592 14 20    16593 3E 62
16594 00 0     16595 28 40
16596 04 4     16597 CD 205
16598 BD 189   16599 07 7
16600 7E 126   16601 4F 79
16602 06 6     16603 00 0
16604 FE 254   16605 21 33
16606 C2 194   16607 F7 247
16608 40 64    16609 2A 42
16610 82 130   16611 40 64
16612 2B 43    16613 7E 126
16614 FE 254   16615 76 118
16616 CA 202   16617 10 16
16618 41 65    16619 7E 126
16620 FE 254   16621 80 128
16622 C2 194   16623 24 36
16624 41 65    16625 22 34
16626 82 130   16627 40 64
16628 C3 195   16629 10 16
16630 41 65    16631 79 121
16632 FE 254   16633 24 36
16634 C2 194   16635 10 16
16636 41 65    16637 2A 42
16638 82 130   16639 40 64
16640 23 35    16641 7E 126
16642 FE 254   16643 76 118
16644 CA 202   16645 10 16
16646 41 65    16647 7E 126
16648 FE 254   16649 80 128
16650 C2 194   16551 24 36
16652 41 65    16653 22 34
16654 82 130   16655 40 64
16656 2A 42    16657 82 130
16658 40 64    16659 7E 126
16660 FE 254   16661 80 128
16662 C2 194   16663 24 36
16664 41 65    16665 36 54
16666 A6 166   16667 2A 42
16668 84 132   16669 40 64
16670 36 54    16671 80 128
16672 01 1     16673 00 0
16674 00 0     16675 C9 201
16676 36 54    16677 80 128
16678 2A 42    16679 84 132
16680 40 64    16681 36 54
16682 80 128   16683 01 1
16684 01 1     16685 00 0
16686 C9 201
```

A disassembly of the machine code part of the listing.

MACHINE DISASSEMBLY

LABELS	ADDRESS	ASSEMBLER	DECIMAL
SET UP	16514	NOP	0
	16515	NOP	0
	16516	NOP	0
	16517	NOP	0
	16518	LD HL,(D-FILE)	42 12 64
	16521	LD DE,512	17 0 2
	16524	ADD HL,DE	25
	16525	LD (16514),HL	34 130 64
SCROLL DOWN	16528	LD HL,(D-FILE)	42 12 64
	16531	LD DE,626	17 114 2
	16534	ADD HL,DE	25
	16535	PUSH HL	229
	16536	LD B,33	6 33
	16538	INC HL	35
	16539	DJ NZ – 1	16 253
	16541	PUSH HL	229
	16542	POP DE	209
	16543	POP HL	225
	16544	LD C,18	14 18
	16546	LD B,33	6 33
	16548	LD A,(HL)	126
	16549	LD(DE),A	18
	16550	DEC DE	27
	16551	DEC HL	43
	16552	DJ NZ – 4	16 250
	16554	DEC C	13
	16555	JR NZ – 7	32 245
SCORE	16557	LD HL,(D-FILE)	42 12 64
	16560	LD DE,12	17 12 0
	16563	ADD HL,DE	25
	16564	LD A,(HL)	126
	16565	INC A	60
	16566	CP 166	254 166
	16568	JR NZ 5	32 5
	16570	LD (HL),156	54 156
	16572	DEC HL	43
	16573	JR 245	24 245
	16575	LD (HL),A	119
MOVE SHIP	16576	LD HL,(16514)	42 130 64
	16579	LD DE,33	17 33 0
	16582	ADD HL,DE	25
	16583	LD (16516),HL	34 132 64
GET KEY	16586	CALL INKEY	205 187 2
	16589	LD B,H	68
	16590	LD C,L	77
	16591	LD D,C	81
	16592	INC D	20
	16593	LD A,O	62 0
	16595	JR Z + 4	40 4
	16597	CALL GET CHAR	205 189 7
	16600	LD A,(HL)	126
	16601	LD C,A	79
	16601	LD B,O	6 0
LEFT	16604	CP 33	254 33
	16606	JP NZ move right	194 247 64
	16609	LD HL,(16514)	42 130 64
	16612	DEC HL	43
	16613	LD A,(HL)	126
	16614	CP 118	254 118
	16616	JP Z print	202 16 65
	16619	LD A,(HL)	126
	16620	CP 128	254 128
	16622	JP NZ crash	194 36 65
	16625	LD (16514),HL	34 130 64
	16628	JP print	195 16 65
	16631	LD A,C	121
RIGHT	16632	CP 36	254 36
	16634	JP NZ print	194 16 65
	16637	LD HL,(16514)	42 130 64
	16640	INC HL	35
	16641	LD A,(HL)	126
	16642	CP 118	254 118
	16644	JP Z print	194 16 65
	16647	LD A,(HL)	126
	16648	CP 128	254 128
	16650	JP NZ crash	194 36 65
	16653	LD (16514),HL	34 130 64
PRINT	16656	LD HL,(16514)	42 130 64
	16659	LD A,(HL)	126
	16660	CP 128	254 128
	16662	JP NZ, crash	194 36 65
	16665	LD(HL),166	54 166
	16667	LD HL,(16516)	42 132 64
	16670	LD (HL),128	54 128
	16672	LD BC,O	1 0 0
	16675	RET	201
CRASH	16676	LD (HL),128	54 128
	16678	LD HL,(16516)	42 132 64
	16681	LD HL,128	54 128
	16683	LD BC,1	1 1 0
	16686	RET	201

Spectramon — part two

Presenting the second part of this feature article, including the full listing of Simon Goodwin's incredible Spectrum monitor program.

Spectramon is written in ZX BASIC but it should be quite easy to convert for other computers. Obviously, it will only be useful on machines which use the Z-80 processor!

The Spectrum CODE function corresponds to ASC on other computers — brackets around its argument are optional in ZX BASIC. String arrays are handled rather oddly by Sinclair BASIC — the variable Z$ is set up by line 40 as having a fixed length of 32. Unused character-positions contain spaces — so that Z$ is simply used as an array of space characters by the instruction formatting routine. The array O$ contains 608 strings (numbered from one, not zero) and each string has a fixed length of nine characters (line 130).

The other string variables are normal 'Microsoft' strings — they vary in length to accommodate whatever is stored in them. ZX BASIC allows substrings to be extracted from a string using the 'TO' instruction — A$(1 TO 1) returns the first character of a string, corresponding to LEFT$(A$,1) in Microsoft BASIC. If A$ is set up as 'SPECTRAMON' then A$,6,3). In short, the 'TO' instruction extracts all the characters from one position TO another, inclusive.

Spectrum BASIC allows long variable-names to be specified, and (unlike Microsoft BASIC) all the characters of a name are significant. On the Spectrum, INDEX and INDIRECT are two different, valid variables — in Microsoft BASIC they will have to be renamed, otherwise they would be treated as the same variable because they have the same first two characters. In some versions of Microsoft BASIC, neither variable name would be allowed since they both contain the key-word 'IN'. Sinclair BASIC is also unusual in that it allows spaces to occur in variable names. Table 1 shows all the variable names used in Spectramon and documents their usage.

Other systems can ignore the lines using COPY to send out a listing and simply LPRINT L$ if LP=1, printing out lines one at a time rather than *en masse*. A user defined function is set up in line 50, but it is fairly easy to code around this if your computer doesn't support that feature. FNH(H$) simply returns the decimal value of the first character in H$ — 1 for '1', 10 for 'A', 11 for 'B' and so on.

Spectramon uses a few PEEKs and POKEs which will not be required on other systems. POKE 23658,8 is a useful command which forces the Spectrum into capitals-lock (selecting a flashing 'C' as a cursor rather than a flashing 'L'). This ensures that commands are entered in capitals (unless the user purposely switches to lower-case in the course of entering a command). The location 23689 contains the number of empty lines on the Spectrum screen — when PEEK 23869 is three or less the screen is assumed to be full since the bottom two lines aren't normally used for text and a line is needed for the 'More? . . .' message.

Location 23560 contains the ASCII code of the key most recently pressed. It is set to 32 when the space bar has been pressed (or is being simulated) and 13 when Enter has been typed.

The last word . . .

When I received my Spectrum I was convinced that I'd never get used to the keyboard. After writing, editing and typing in Spectramon I was well-practiced! Hopefully, the program also illustrates a few useful quirks of ZX BASIC, both from the BASIC and the assembler programmer's point of view.

Z$ —	Fixed length string of 32 spaces, used in formatting.
H$ —	Hex characters '0'-'F' — also a local variable used in the Hex-Decimal conversion function, FN H (line 50).
LP —	'Flag' set to 1 if printout is required.
DEC —	'Flag' set to 1 if numbers must be output in decimal.
CHECK INDEX —	Line number of the routine which checks to see whether an operation could involve IX or IY.
GET INSTRUCTION —	Line number of the routine which formats a complete line of disassembler output.
MAKE TEXT —	Line number of the routine which formats a complete line of disassembler output.
BYTE VALUE —	Line number of a routine which expresses the contents of C (0-255) in C$, using the current base.
WORD VALUE —	Line number of a routine which sets up C$ with a string copy of C (0-65535) in the current base.
F$ —	String containing register names.
O$ —	String array containing the opcode text.
I,K,T —	Loop counters and temporary values.
A$ —	The command typed in by the user.
C$ —	The first character of the command.
SUB —	Line number of the chosen monitor subroutine.
LOC —	The location being examined by the monitor.
L$ —	The line of text to be output by the monitor.
I0,I1,I2 —	The instruction code and its operands.
N$ —	The name of the current index register.
S$ —	The name of the current indirect register (N$).
M$ —	The mnemonic form of the instruction.
NBYTES —	Length of instruction, in bytes.
INDEX —	Set to 1 if IX or IY are to replace HL.
INDIRECT —	Set to 1 if (IX) or (IY) are to replace (HL).
R$ —	Character within instruction mnemonic.
MODE —	Addressing mode 0-9; declares number and format of operands.
C —	Number for conversion into a decimal or Hex string.
C$ —	Number after conversion into a string.
D$ —	Part of disassembler output line.

Table 1. Variables used in Spectramon.

```
  40 DIM Z$(32). REM Spaces
  50 DEF FN H(H$)=CODE H$-48-7*(
H$>"@"): REM Convert hex charact
er to decimal
  60 LET LP=0: REM No printer
  70 LET DEC=0: REM Hex output
  80 LET CHECK INDEX=1600
  90 LET GET INSTRUCTION=2000
 100 LET MAKE TEXT=4000
 110 LET BYTE VALUE=5000
 120 LET WORD VALUE=5020
 130 DIM O$(608,9): REM Opcodes
 140 RESTORE
 150 READ F$
 160 LET I=1: REM Opcode No.
```

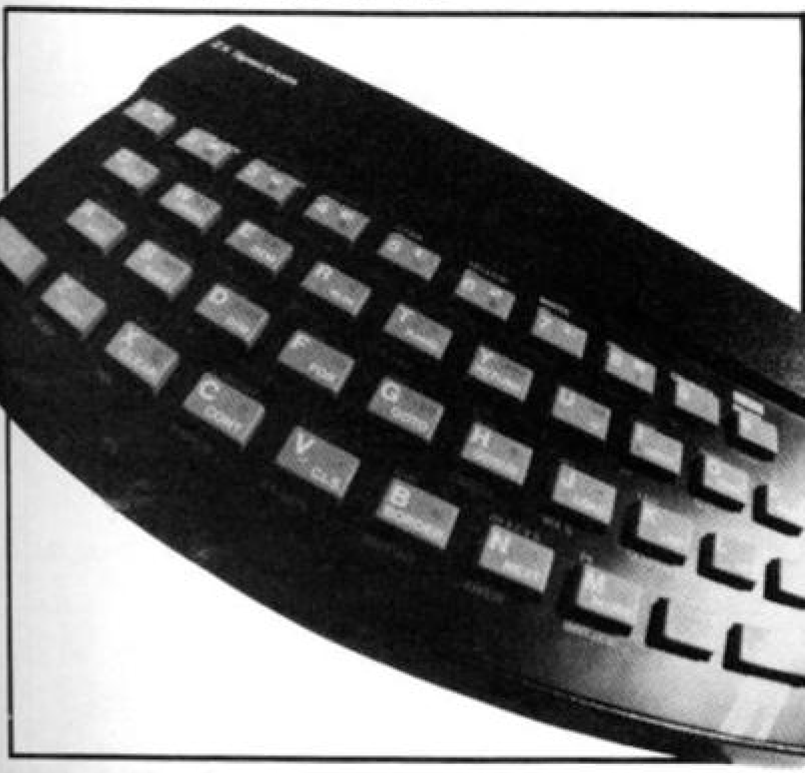
```
170 READ O$(I)
180 IF O$(I,1 TO 1)="1" THEN FOR J=1 TO 7: LET O$(I+J)=O$(I): NEXT J: LET I=I+7: REM Standard 8-bit operand
190 LET I=I+1: IF I<609 THEN GO TO 170: REM Repeat for all opcodes

195 PRINT '
200 LET H$="0123456789ABCDEF": REM Hex characters
290 REM ** Command Menu
300 CLS : PRINT "COMMANDS"
310 PRINT '"Q Return to ZX BASIC."
320 PRINT '"D<address> Disassemble Program."
330 PRINT '"N<address> Numeric dump memory"
340 PRINT '"A<address> ASCII mem. display."
350 PRINT '"P Printer option (now ";: IF LP THEN PRINT "ON)"
355 IF LP=0 THEN PRINT "OFF)"
360 PRINT '"B Base Selection (now ";: IF DEC THEN PRINT "DEC)"
365 IF DEC=0 THEN PRINT "HEX)"
390 POKE 23658,8: REM Force caps lock
400 INPUT "Command?"; LINE A$
410 IF LEN A$<1 THEN GO TO 400: REM Nothing typed
415 LET C$=A$(1 TO 1): LET SUB=0: IF C$="Q" THEN CLS : STOP
420 IF C$="B" THEN LET DEC=1-DEC: GO TO 200: REM Reverse value of DEC - Base flag
425 IF C$="P" THEN LET LP=1-LP: GO TO 200: REM Reverse value of LP - Printer flag
430 IF LEN A$<2 THEN GO TO 400: REM No address specified
440 LET A$=A$(2 TO LEN A$): IF A$(LEN A$ TO LEN A$)="H" THEN GO TO 2200: REM Hex address?
450 FOR I=1 TO LEN A$: IF A$(I TO I)<"0" OR A$(I TO I)>"9" THEN GO TO 400: REM Reject if address is not numeric!
460 NEXT I: LET LOC=VAL A$: REM Set up start address
490 REM ** Select subroutine
500 IF C$="D" THEN LET SUB=1000
510 IF C$="A" THEN LET SUB=7000
520 IF C$="N" THEN LET SUB=7500
590 IF SUB=0 OR LOC>65535 THEN GO TO 400: REM Ignore silly address or command
595 CLS : POKE 23560,0: REM Clear BASIC key buffer
600 GO SUB SUB: PRINT L$
605 IF LOC>65535 THEN PRINT '"End of memory.": POKE 23560,32: GO TO 610: REM Pretend SPACE was typed
606 IF PEEK 23689<4 AND LP THEN COPY : REM Screen Full & print needed
607 IF PEEK 23689<4 AND LP=0 THEN PRINT "More? (Enter = NO)": POKE 23560,32: GO TO 610: REM Screen Full - pause
610 IF PEEK 23560=32 THEN GO TO 610: REM Wait until key<>SPACE
612 IF PEEK 23689<3 THEN CLS : REM Start a new screen
615 IF PEEK 23560<>13 THEN GO TO 600: REM Do another line unless ENTER was typed
620 GO TO 200: REM Return to menu
1000 LET I0=PEEK LOC: REM Get opcode number
1010 IF I0=203 THEN GO TO 1200: REM Bitwise operation (Z80)
1020 IF I0=237 THEN GO TO 1300: REM Miscellaneous Z80 operation
1040 IF I0=221 THEN GO TO 1500: REM Use IX instead of HL
1050 IF I0=253 THEN GO TO 1520: REM Use IY instead of HL
1060 LET I1=PEEK (LOC+1): REM Get LSB of operand
1070 LET I2=PEEK (LOC+2): REM Get MSB of operand (if any!)
1080 GO SUB GET INSTRUCTION
1090 LET N$="HL": LET S$="(HL)"
1100 GO SUB CHECK INDEX: GO SUB MAKE TEXT
1110 LET L$=L$+M$: REM Add instruction to line
1120 RETURN
1190 REM ** CB codes
1200 LET I0=PEEK (LOC+1)+256: REM Compute bitwise inst. number
1210 GO SUB GET INSTRUCTION
1220 IF M$="?" THEN GO SUB MAKE TEXT: GO TO 1110: REM Got instruction
1230 LET NBYTES=2
1240 GO TO 1090: REM Finished
1290 REM ** ED codes
1300 LET I0=PEEK (LOC+1): REM Get the real inst. number
1310 IF I0<64 OR (I0>127 AND I0<160) OR I0>191 THEN LET I0=191: REM Not a valid ED code
1315 IF I0<128 THEN LET I0=I0+32: REM Work out position in O$
1320 LET I0=I0+417
1330 LET I1=PEEK (LOC+2): LET I2=PEEK (LOC+3): REM Find new operand
1340 GO SUB GET INSTRUCTION
1350 IF M$="?" THEN GO SUB MAKE TEXT: GO TO 1110
1360 LET NBYTES=NBYTES+1: REM Longer than expected
1370 GO TO 1090
1490 REM ** DD codes
1500 LET N$="IX": REM Use IX instead of HL
1510 GO TO 1530: REM Continue as for IY
1520 LET N$="IY": REM FD codes
1530 LET C=PEEK (LOC+2): REM Get offset for (IY+offset)
1540 IF C=0 THEN LET S$="("+N$+")"
1550 IF C>0 AND C<128 THEN GO SUB BYTE VALUE: LET S$="("+N$+"+"+C$+")"
1555 IF C>127 THEN LET C=256-C: GO SUB BYTE VALUE: LET S$="("+N$+"-"+C$+")"
1560 IF PEEK (LOC+1)=203 THEN GO TO 1660: REM Indexed bit operation
1570 LET I0=PEEK (LOC+1): LET I1=PEEK (LOC+2): LET I2=PEEK (LOC+3): REM Get the equivalent HL inst.
1580 IF I0=54 THEN LET I2=0: LET I1=PEEK (LOC+3): REM 8 bit indirect load - special case
1590 GO SUB GET INSTRUCTION
1600 LET INDEX=0: LET INDIRECT=0: REM Not using IX or IY
1610 IF M$<>"?" THEN GO SUB CHECK INDEX
1620 LET NBYTES=NBYTES+INDIRECT+INDEX: REM Adjust instruction length
1625 IF INDEX=0 THEN LET NBYTES=NBYTES+INDIRECT: REM Extra byte needed for offset
1630 GO SUB MAKE TEXT
1640 GO TO 1110
```

```
1660 LET I0=PEEK (LOC+3)+256: RE
M 8 bit indirect bitwise inst.
1670 GO SUB GET INSTRUCTION
1680 LET INDEX=0: LET INDIRECT=0
: REM Pretend IX & IY aren't nee
ded
1690 IF M$<>"?" THEN GO SUB CHEC
K INDEX: REM Make sure
1700 LET NBYTES=NBYTES+3*INDIREC
T
1710 GO SUB MAKE TEXT
1720 GO TO 1110
1800 LET INDIRECT=0: LET INDEX=0
: LET I=5
1810 LET I=I+1: IF I>LEN M$ THEN
 RETURN
1820 LET R$=M$(I TO I): IF R$<>"
#" AND R$<>"*" THEN GO TO 1810:
REM Check for abbreviations
1330 IF R$="*" THEN GO TO 1880
1340 LET INDEX=1: REM use IX/IY
1850 LET M$=M$(1 TO I-1)+N$+M$(I
+1 TO LEN (M$)): REM Modify inst
ruction
1860 RETURN
1680 LET INDIRECT=1: REM Use (IX
) or (IY) instead of (HL)
1890 LET M$=M$(1 TO I-1)+S$+M$(I
+1 TO LEN (M$)): REM Insert regi
ster name
1900 RETURN
2000 LET I$=U$(I0+1): REM Get te
xt of opcode
2003 IF I$(LEN I$)=" " THEN LET
I$=I$(1 TO LEN I$-1): GO TO 2003
: REM Discard trailing spaces
2005 LET MODE=CODE I$-48: REM Ge
t addressing mode (if special)
2010 IF MODE<1 OR MODE>9 THEN LE
T MODE=0: GO TO 2020: REM Not sp
ecial
2015 LET I$=I$(2 TO LEN I$): REM
 Strip off the 'mode' marker
2020 FOR I=1 TO LEN I$: IF I$(I
TO I)=" " THEN GO TO 2045: REM F
ind operand field
2025 NEXT I
2030 LET A$=I$+Z$(1 TO 5-LEN I$)
: REM no operand, format neatly
2035 LET B$="": REM Operand!
2040 GO TO 2055
2045 LET A$=I$(1 TO I)+Z$(1 TO 5
-I): REM This is the opcode
2050 LET B$=I$(I+1 TO LEN I$): R
EM and this is the operand
2055 GO TO 3000+MODE*100: REM Pr
ocess in accordance with addr mo
de
2190 REM ** Convert hex to dec
2200 IF LEN A$<2 THEN GO TO 400:
 REM Must be at least 1 digit
2210 LET A$="000"+A$( TO LEN A$-
1): LET A$=A$(LEN A$-3 TO LEN A$
): REM Force 4 character field
2220 LET LOC=0: FOR I=0 TO 3: LE
T LOC=LOC+FN H(A$(LEN A$-I))*16↑
I: NEXT I: GO TO 500: REM Get va
l in LOC
2990 REM ** Use addressing mode
3000 LET NBYTES=1: REM Implied a
ddressing - no operand bytes
3010 LET M$=A$+B$: REM Build ent
ire instruction text
3020 RETURN
3100 LET NBYTES=1: REM 8 bit ope
rand in or pointed to by registe
rs
3110 IF LEN (B$)<>0 THEN LET B$=
B$+",": REM Format neatly
3115 LET K=I0-INT (I0/8)*8+1: RE
M Get operand number from inst.
3120 LET M$=A$+B$+F$(K): REM Add
 appropriate symbol
3130 IF I0=118 THEN LET M$="HALT
": REM LD (HL),(HL) doesn't exis
t - HALT takes its place
3140 RETURN
3200 LET NBYTES=2: REM 8 bit num
ber is operand
3210 IF LEN (B$)<>0 THEN LET B$=
B$+","
3220 LET C=I1: REM Get number
3230 GO SUB BYTE VALUE
3240 LET M$=A$+B$+C$: REM Build
inst.
3250 RETURN
3300 LET NBYTES=3: REM 16 bit nu
mber is operand
3310 IF LEN (B$)<>0 THEN LET B$=
B$+","
3320 LET C=256*I2+I1
3330 GO SUB WORD VALUE
3340 LET M$=A$+B$+C$
3350 RETURN
3400 LET NBYTES=2: REM Relative
jump offset is operand
3410 IF LEN (B$)<>0 THEN LET B$=
B$+","
3420 LET C=LOC-254+I1: REM Work
out absolute addr. from offset
3430 IF I1<128 THEN LET C=C+256:
 REM Can go back or forward
3440 GO TO 3330: REM Display abs
olute address of target inst.
3500 LET NBYTES=2: REM Port numb
er 0-255 is in operand byte
3510 LET C=I1
3520 GO SUB BYTE VALUE
3530 GO TO 3630

3600 LET NBYTES=3: REM Indirect
address is in operand
3610 LET C=256*I2+I1
3620 GO SUB WORD VALUE
3630 IF LEN (B$)<>0 THEN LET B$=
B$+","
3640 LET M$=A$+B$+"("+C$+")": RE
M Brackets indicate indirection
3650 RETURN
3700 LET NBYTES=2: REM IN to por
t numbered by operand
3710 LET C=I1
3720 GO SUB BYTE VALUE
3730 GO TO 3830
3800 LET NBYTES=3: REM STORE at
operand address
3810 LET C=256*I2+I1
3820 GO SUB WORD VALUE
3830 IF LEN B$<>0 THEN LET B$=",
"+B$
3840 LET M$=A$+"("+C$+")"+B$: RE
M Brackets indicate address not
value
3850 RETURN
3900 LET NBYTES=1: REM Peculiar
code
3910 LET M$="?"
3920 RETURN
3990 REM ** Build up o/p line
4000 LET C=LOC: GO SUB WORD VALU
E
4010 IF DEC THEN LET L$=C$+" ":
GO TO 4030
4020 LET L$=C$(1 TO 4)+" ": REM
Remove 'H' to shorten line
4030 LET D$="": FOR T=LOC TO LOC
+NBYTES-1
4070 LET C=PEEK T: REM Get data
4075 IF DEC THEN LET DEC=0: GO S
UB BYTE VALUE: LET DEC=1: GO TO
4090: REM Always use hex (neater
)
4080 GO SUB BYTE VALUE
4090 LET D$=D$+C$(1 TO 2): REM N
o 'H' here either
4100 NEXT T: REM Process all the
 bytes of the instruction
4110 LET L$=L$+D$+Z$(1 TO 2*(4-N
BYTES))+" ": REM Add spaces tidily
```

```
4120 LET LOC=LOC+NBYTES: REM Adv
ance to next instruction
4130 RETURN
4990 REM ** Convert C to C$
5000 IF DEC THEN LET C$=STR$ C:
RETURN : REM Decimal is easy
5010 LET C$=H$(C/16+.5 TO C/16+.
5): LET C=C-INT (C/16)*16+.5: LE
T C$=C$+H$(C TO C)+"H": RETURN :
 REM 8 bit hex value
5020 IF DEC THEN LET C$=STR$ C:
RETURN
5025 LET CT=C/256: LET C$=H$(CT/
16+.5 TO CT/16+.5): LET CT=CT-IN
T (CT/16)*16+.5: LET C$=C$+H$(CT
 TO CT)
5030 LET CT=C-INT (C/256)*256: L
ET C$=C$+H$(CT/16+.5 TO CT/16+.5
): LET CT=CT-INT (CT/16)*16+.5:
LET C$=C$+H$(CT TO CT)+"H": RETU
RN : REM 16 bit hex value
5990 REM ** Z80 Instructions
6000 DATA "BCDEHL*A"
6010 DATA "NOP","3LD BC","LD (BC
),A","INC BC","INC B","DEC B","2
LD B","RLCA","EX AF,AF'","ADD #,
BC","LD A,(BC)","DEC BC","INC C"
,"DEC C","2LD C","RRCA","4DJNZ E
","3LD DE","LD (DE),A","INC DE"
6020 DATA "INC D","DEC D","2LD D
","RLA","4JR","ADD #,DE","LD A,(
DE)","DEC DE","INC E","DEC E","2
LD E","RRA","4JR NZ","3LD #","6L
D #","INC #","INC H","DEC H","2L
D H","DAA","4JR Z","ADD #,#"
6030 DATA "6LD #","DEC #","INC L
","DEC L","2LD L","CPL","4JR NC"
,"3LD SP","6LD A","INC SP","INC
*","DEC *","2LD *","SCF","4JR C"
,"ADD #,SP","6LD A","DEC SP","IN
C A","DEC A","2LD A","CCF"
6040 DATA "1LD B","1LD C","1LD D
","1LD E","1LD H","1LD L","1LD *
","1LD A","1ADD A","1ADC A","1SU
B A","1SBC A","1AND","1XOR","1OR
","1CP","RET NZ","POP BC","3JP N
Z","3JP","3CALL NZ"
```

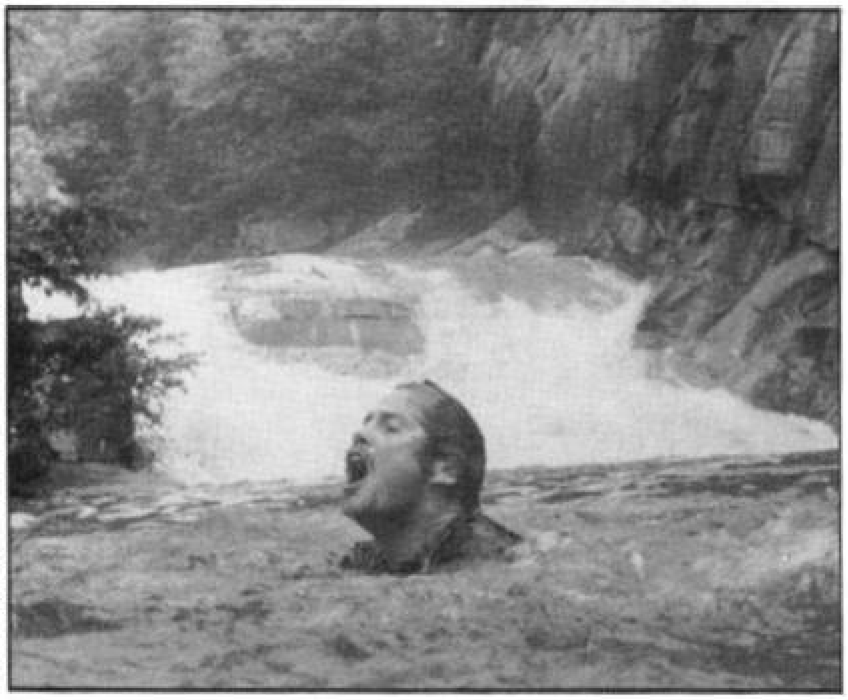


```
6050 DATA "PUSH BC","2ADD A","RS
T 0","RET Z","RET","3JP Z","9","
3CALL Z","3CALL","2ADC A","RST 8
H","RET NC","POP DE","3JP NC","7
OUT A","3CALL NC","PUSH DE","2SU
B A"
6060 DATA "RST 10H","RET C","EXX
","3JP C","5IN A","3CALL C","9",
"2SBC A","RST 18H","RET PO","POP
 #","3JP PO","EX (SP),#","3CALL
PO","PUSH #","2AND","RST 20H"
6065 DATA "RET PE","JP (#)","6JP
 PE","EX DE,HL","3CALL PE","9","
2XOR","RST 28H","RET P","POP AF"
,"3JP PE","DI","3CALL P"
6070 DATA "PUSH AF","2OR","RST 3
0H","RET M","LD SP,#","3JP M","E
I","3CALL M","9","2CP","RST 38H"
6080 REM Z80 CB codes
6090 DATA "1RLC","1RRC","1RL","1
RR","1SLA","1SRA","9","9","9","9
","9","9","9","9","9","1SRL"
6100 DATA "1BIT 0","1BIT 1","1BI
T 2","1BIT 3"
6110 DATA "1BIT 4","1BIT 5","1BI
T 6","1BIT 7"
6120 DATA "1RES 0","1RES 1","1RE
S 2","1RES 3"
6130 DATA "1RES 4","1RES 5","1RE
S 6","1RES 7"
6140 DATA "1SET 0","1SET 1","1SE
T 2","1SET 3"
6150 DATA "1SET 4","1SET 5","1SE
T 6","1SET 7"
6160 REM Z80 ED codes (40-7F)
6170 DATA "IN B,(C)","OUT (C),B"
,"SBC HL,BC","8LD BC","NEG","RET
N","IM 0","LD I,A"
6180 DATA "IN C,(C)","OUT (C),C"
,"ADC HL,BC","6LD BC","9","RETI"
,"9","LD R,A"
6190 DATA "IN D,(C)","OUT (C),D"
,"SBC HL,DE","8LD DE","9","9","I
M 1","LD A,I"
6200 DATA "IN E,(C)","OUT (C),E"
,"ADC HL,DE","6LD DE","9","9","I
M 2","LD A,R"
6210 DATA "IN H,(C)","OUT (C),H"
,"SBC HL,HL","8LD HL","9","9","9
","RRD"
6220 DATA "IN L,(C)","OUT (C),L"
,"ADC HL,HL","6LD HL","9","9","9
","RLD"

6230 DATA "9","9","SBC HL,SP","8
LD SP","9","9","9","9","IN A,(C)
","OUT (C),A","ADC HL,SP","6LD S
P","9","9","9","9"
6240 REM More ED codes (A0-BF)
6250 DATA "LDI","CPI","INI","OUT
I","9","9","9","9"
6260 DATA "LDD","CPD","IND","OUT
D","9","9","9","9"
6270 DATA "LDIR","CPIR","INIR","
OTIR","9","9","9","9"
6280 DATA "LDDR","CPDR","INDR","
OTDR","9","9","9","9"
6990 REM ** Character dump
7000 LET C=LOC: GO SUB 5020: LET
 L$=C$+" ": FOR C=0 TO 15: IF C+
LOC>65535 THEN LET I0=32: GO TO
7030: REM Beware end of RAM
7010 LET I0=PEEK (C+LOC): IF I0>
127 THEN LET I0=I0-128: REM Stri
p 8th bit (parity or flag)
7020 IF I0<32 THEN LET I0=46: RE
M Print controls as dots
7030 LET L$=L$+CHR$ I0
7040 NEXT C: LET LOC=LOC+16: RET
URN
7490 REM ** Numeric dump
7500 LET C=LOC: GO SUB WORD VALU
E: LET L$=C$+" ": FOR I=0 TO 7:
IF LOC+I>65535 THEN LET I0=0: GO
 TO 7520: REM Beware end of RAM
7510 LET I0=PEEK (LOC+I)
7520 LET C=I0: GO SUB 5000: IF D
EC THEN LET L$=L$+" "+C$
7525 IF DEC=0 THEN LET L$=L$+" "
+C$(1 TO 2): REM Strip 'H' to ke
ep under 32 columns
7530 NEXT I: LET LOC=LOC+8: RETU
RN
```

River of death

A Reynolds of Chatham invites you to try jumping across the river of death . . .



Photograph courtesy of Warner Bros. Inc.

You are a frog and your destiny is to try and jump your way across a busy road and then try and cross the river.

In a similar way to the arcade game called Frogger, you move your frog across the road trying very hard not to get hit by one of the moving cars. Once across the road, you will find safety on the river bank. The next stage of the game is to cross the river, which is populated by turtles and floating logs. You must get your frog to jump across the river via the logs and turtles, but like all games of this kind that's easier said than done!

Jump to it

Once you get to the other side of the river, you have to jump your frog from the turtle or log you are on to the gap in the opposite bank. Movement is made via the '5' key to go left, the '8' key to move right and the '7' key to go forward. Points are gained along the way; you get 10 points for each movement forward and 100 points for reaching the river bank. But remember there's no turning back!

```
   1 LET HI=0
   2 GOTO 1000
   3 LET SC=0
   4 LET LIVES=3
   6 PRINT AT 1,3;" SCREEN ONE "
;AT 2,1;" T H E   T R A F F I C
 "
   7 PRINT AT 20,0;"SCORE DISPLA
YED AFTER COMPLETIONOF THIS STAG
E"
   9 PRINT AT 18,0;"LIVES";LIVES
;TAB 10;"SCORE";SC;TAB 20;"HI-SC
ORE";HI
  10 LET ST=1
  13 PRINT AT 4,0;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
  15 LET E=15
  16 LET F=15
  17 LET P=CODE "■"
  20 PRINT AT 15,0;"▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
  30 LET A$="CCCC  CCCCC     CCC
 CCC  CCCC   "
  31 LET A$=A$+A$
  32 LET C$=" CCCCCC    CCCCC   CC
C  CCC    CCC    "
  33 LET C$=C$+C$
  34 LET B$="   CCCC    CCC   CCC
CCCCCC  CC    "
  35 LET B$=B$+B$
  36 LET D$="  CCCC   CCCC      CCC
C   CCCC   CC   "
  37 LET D$=D$+D$
  40 LET X=1
  41 LET A=65
  42 LET B=35
  50 LET Y=31
  70 LET X=X+1
  71 LET A=A-1
  72 LET B=B-1
  80 LET Y=Y+1
  83 PRINT AT E,F;CHR$ P
  85 PRINT AT 7,0;A$(X TO Y);AT
11,0;C$(X TO Y);AT 9,0;B$(B TO A
);AT 13,0;D$(B TO A)
  90 IF X=34 THEN GOTO 40
 100 LET F=F+(INKEY$="8")-(INKEY
$="5")
 110 LET M$=INKEY$
 120 IF M$="7" THEN LET E=E-2
 122 IF M$="7" THEN LET SC=SC+10
 125 PRINT AT E,F;
 130 LET P=PEEK (PEEK 16398+256*
PEEK 16399)
 135 PRINT AT E,F;"■"
 140 IF P=CODE "■" THEN GOTO 150
 145 IF P=CODE "▒" THEN GOTO 200
 147 GOTO 70
 150 FOR I=1 TO 10
 155 NEXT I
 157 PRINT AT E,F;"■"
 158 FOR I=1 TO 10
 159 NEXT I
 160 PRINT AT E,F;"X"
 165 FOR I=1 TO 10
 166 NEXT I
 167 PRINT AT E,F;"....."
 168 FOR I=1 TO 10
 169 NEXT I
 170 PRINT AT E,F;("SPLAT" AND S
T=1)+("SPLOSH" AND ST=2)+("CRUNC
H" AND ST=3)
 171 LET LIVES=LIVES-1
 172 FOR I=1 TO 20
 173 NEXT I
 176 IF LIVES=0 THEN GOTO 181
 179 CLS
 180 GOTO 5
 181 PRINT AT 10,10;"GAME OVER"
 182 FOR I=1 TO 50
 183 NEXT I
 184 CLS
 185 PRINT "YOU FINAL SCORE = ";
SC
 186 IF SC>HI THEN LET HI=SC
 188 PRINT "AGAIN (Y/N) "
 189 IF INKEY$="Y" THEN GOTO 2
 190 IF INKEY$="N" THEN GOTO 200
0
 191 GOTO 189
 200 FOR I=1 TO 50
 210 NEXT I
 220 CLS
 230 FOR I=0 TO 20
 240 PRINT AT I,0;"WELL DONE YOU
 HAVE MADE IT......"
```

```
 245 PRINT AT I+1,0;"WELL DONE Y
OU HAVE MADE IT ....."
 250 NEXT I
 260 PRINT AT 21,0;"WELL DONE YO
U HAVE MADE IT ...."
 266 PRINT AT 11,4;"SCORE SO FAR
=";SC
 270 FOR I=1 TO 100
 280 NEXT I
 290 CLS
 310 PRINT AT 1,3;" SCREEN TWO "
;AT 2,1;" T H E   R I V E R "
 320 PRINT AT 4,0;"
                 ";Z$
 330 PRINT AT 15,0;"

"
 340 PRINT AT 18,0;"LIVES";LIVES
;TAB 10;"SCORE";SC;TAB 20;"HI-SC
ORE";HI
 350 PRINT AT 20,0;"SCORE DISPLA
YED AFTER COMPLETIONOF THIS STAG
E"
 360 LET A$="
       "
 370 LET A$=A$+A$
 380 LET B$="
       "
 390 LET B$=B$+B$
 400 LET C$="
       "
 410 LET C$=C$+C$
 412 LET E=15
```



```
 414 LET F=15
 416 LET P=CODE " "
 420 LET D$="
       "
 425 LET D$=D$+D$
 430 LET X=1
 431 LET Y=31
 432 LET A=63
 434 LET B=33
 440 LET X=X+1
 442 LET Y=Y+1
 444 LET A=A-1
 446 LET B=B-1
 448 PRINT AT E,F;CHR$ P
 450 PRINT AT 7,0;A$(B TO A);AT
9,0;B$(X TO Y);AT 11,0;C$(B TO A
);AT 13,0;D$(X TO Y)
 460 IF X=32 THEN GOTO 430
 462 LET F=F+(INKEY$="8")-(INKEY
$="5")
 463 LET F=F+(E=11)+(E=7)-(E=13)
-(E=9)
 464 LET M$=INKEY$
 465 IF M$="7" THEN LET E=E-2
 466 IF M$="7" THEN LET SC=SC+10
 469 PRINT AT E,F;
 470 LET P=PEEK (PEEK 16398+256*
PEEK 16399)
 480 PRINT AT E,F;" "
 490 IF P=0 THEN GOTO 550
 495 IF E=5 THEN GOTO 550
 500 GOTO 440
 550 IF E=5 AND P=0 THEN GOTO 57
0
 555 IF E=5 AND P<>0 THEN LET ST
=3
 560 IF P=0 AND E<>5 THEN LET ST
=2
 565 GOTO 150
 570 FOR I=6 TO 30 STEP 6
 580 PRINT AT 5,I-1;
 590 LET P=PEEK (PEEK 16398+256*
PEEK 16399)
 600 IF P=CODE " " THEN LET Z$(I
)=" "
 610 NEXT I
 613 IF Z$="
                 " THEN GOTO 700
 620 FOR I=1 TO 10
 621 NEXT I


 630 CLS
 640 PRINT AT 4,3;"WELL DONE";TA
B 2;"100 POINTS BONUS"
 650 LET SC=SC+100
 660 PRINT "SCORE NOW = ";SC
 670 FOR I=1 TO 50
 680 NEXT I
 685 CLS
 690 GOTO 5
 700 FOR I=1 TO 10
 701 NEXT I
 702 CLS
 703 LET SC=SC+100
 704 PRINT ,,,,,,"WELL DONE ALL
5 ","FROGS HOME ",,"SCORE NOW =
";SC
 710 LET Z$="
        "
 720 FOR I=1 TO 50
 721 NEXT I
 730 CLS
 740 GOTO 5
1000 REM
1005 SLOW
1010 LET Z$="
        "
1015 IF INKEY$<>"" THEN GOTO 101
5
1020 PRINT "DO YOU WANT INSTRUCT
IONS ?"
1025 LET N$=INKEY$
1030 IF N$="" THEN GOTO 1025
1035 CLS
1040 IF N$="N" THEN GOTO 3
1050 CLS
1070 PRINT "    YOUR FROG........
                THE CARS.........
                THE LOGS.........
                THE TURTLES......

"
1080 PRINT ,,TAB 10;"SCORE TABLE
","10 PTS PER FORWARD MOVE","100
 PTS FOR REACHING RIVER BANK",,T
AB 10;"KEYS"
1090 PRINT "          5.........LEFT
                      8........RIGHT
                      7..........UP
"
1100 PRINT ,,"THE OBJECT OF THE
GAME IS","TO CROSS THE ROAD BY G
OING","THROUGH THE GAPS BETWEEN
THE","CARS. THEN YOU MUST GO ACR
OSS","THE RIVER BY JUMPING ON TH
E","LOGS AND TURTLES AND INTO A"
,"GAP IN THE RIVER BANK"
1110 IF INKEY$="" THEN GOTO 1110
1115 CLS
1120 GOTO 3
2000 IF INKEY$<>"" THEN GOTO 200
0
2005 PRINT "OH PLEASE (Y/N)"
2010 IF INKEY$="Y" THEN GOTO 2
2020 IF INKEY$="N" THEN GOTO 203
0
2025 GOTO 2010
2030 PRINT "OK, IF THATS WHAT YO
U WANT."
2032 FOR I=1 TO 50
2033 NEXT I
2040 CLS
2050 STOP
9000 SAVE "F"
9010 RUN
```

A stitch in time?

Nick Pearce investigates a couple of software packages for your ZX81.



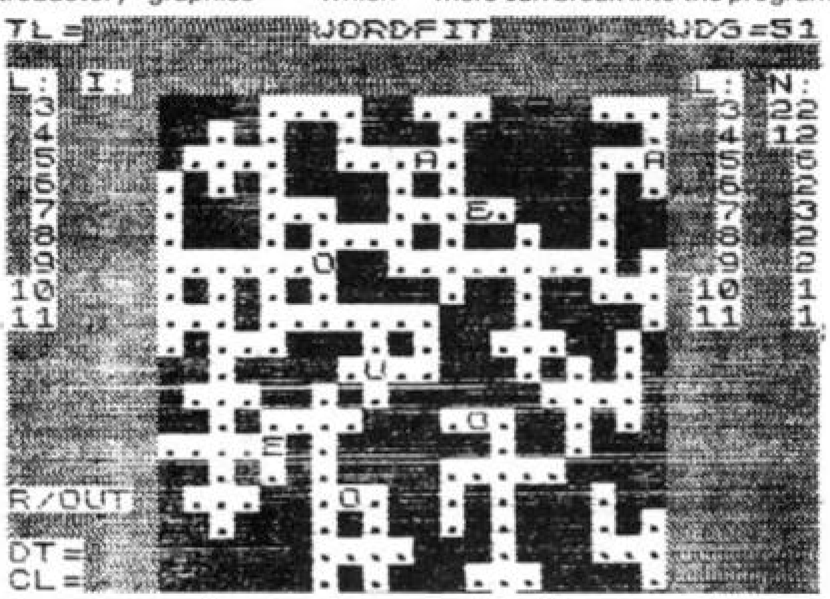
Photograph courtesy of The Rank Organisation.

The First Aid Program — Network Computer Systems Ltd

The First Aid Program is a short instruction course which aims to teach the user how to recognise danger signs and the immediate actions to take for suffocation, bleeding, poisoning, burns and scalds, broken bones and shock.

LOADing the program, which is written in BASIC, takes about six minutes. After some nice introductory graphics — which mechanisms, emergency procedures, broken bones, bleeding and shock. Each time you give a wrong, or not quite right, answer to a question, the section of the program dealing with that subject is repeated. A rather tedious process, but one which should drive home the important points.

It is interesting to see the way Network Computer Systems have applied computer instruction methods to the ZX81. BASIC, rather than machine code, is indeed an appropriate language — budding programmers can break into the program



A sample screen dump from the program, Wordfit.

are a feature throughout this program — a menu is displayed from which you can opt to study, revise a subject, or test yourself. The study course takes about 15 minutes. It is quite short in terms of the quantity of material included, and instruction is by way of short explanatory notes (rather simplistic at times) centred around a diagram of the circulatory system. This is a good idea which works well and could be developed further. Blood flow is shown through heart, lungs and tissues and the effect, for example, of suffocation on blood flow is shown. Users can opt to revise or test theirselves on any of the eight subjects included in the study course, including body and modify it as they see fit, learning something of computer instruction and First Aid into the bargain.

Someone who wants to learn First Aid — and we should all prepare ourselves for emergencies — should do a proper First Aid course which includes both formal instruction and practical training. All this computer program does is to summarise some of the main principles of saving life. A 16K ZX81 program could not, of course, be expected to do more than scratch the surface of such an extensive subject — the authorised First Aid manual of the St John Ambulance Association runs to over 200 pages!

If you want to learn First Aid, enroll with your local First Aid Centre, such as the St John Ambulance Brigade or the Red Cross. However, I see the usefulness of this program as an introduction to First Aid for ZX81 hobbyists, awakening an interest in an important subject they might otherwise have avoided. A well thought-out program which uses the ZX81 graphics to good effect, although short in content and a little simplistic in its approach.

The First Aid Program costs £4.99p and is available from Network Computer Systems Ltd, 39 Bampton Road, Luton, Beds LU4 ODD.

Wordfit — RAM Writer

Wordfit is a crossword type game — you do not have clues, but fit words into a randomly designed grid. Not as easy as it sounds!

The program is written in machine code. To start, you select one of the eight game options: four grid sizes (8x8, 12x12, 16x16 or 20x20) each with or without a random insertion of vowels. The pattern within the grid is set at random and changes from game to game.

All you have to do is fill up the rows and columns on the grid with interlocking words. The program will not allow you to use the same word twice, and only letters will be accepted — hyphens, for example, are not allowed. Operation of the program and entry of letters is facilitated by single key commands: '9' deletes the last letter inserted, '1' clears the whole square, and the cursor keys move you around the grid, for example. This system works very well, although the program crashes if you press Shift and another key. You can COPY the screen onto a printer.

As well as the grid, a lot of other information is displayed on the screen; total number of words inserted, number of word spaces on the grid, word lengths, etc. A game which is interesting and fun to play, and educational too. You are invariably left with a few rows or columns into which no word will seem to fit, which can be frustrating. I suppose this program could be used to design one's own crossword puzzle, although you do have to start with a random grid pattern.

Wordfit costs £5.00 and is available from RAM Writer, 3 Vumba House, 2 Cedar Gardens, Sutton, Surrey SM2 5DB.

Biorhyt calcula



According to some, each of us are governed by three life cycles which are put into motion from the moment of our birth. Thus, from the date of your birth, it should be possible to calculate just how your life cycles are on any particular date.

The three life cycles on which these programs are based are as follows:

Physical cycle — This is a 23 day life cycle relating to your physical status.
Emotional cycle — This is a 28 day cycle and relates to your emotional state at any time in the month.
Mental cycle — This is a 33 day cycle pertaining to your mental prowess.

Using this data, Phil began writing a 1K program for the ZX81 which would indicate at what stage each of these cycles were at, once the user had input his or her date of birth.

In this short program, the first input is made at line 10 where your birthdate should be entered in the form DD MM YYYY, ie 18 02 1957. The second input, at line 20, should be the date you would like your biorhythms calculated for. This should be entered in exactly the same format as the input required in line 10.

The output of the program is the number of days before you reach the 'worst' point in that cycle. Thus, the result will be a number for each cycle — that is, the number of days until you reach the critical stage (0 being 'today').

Rhythm 'n' blues

The 1K can easily be expanded, and that's just what Phil Lester has done. His second approach to writing a program to calculate biorhythms takes up 3K of memory in the ZX81, but is similar in structure to the 1K program. This second program is also a good example of making a program user friendly.

So, if you think you're going to have the 'blues' on a certain day, why not check up on your rhythms first.

thm
tors

How are you feeling today? Phil Lester of Hemel Hempstead will tell you with these two programs for the ZX81.

```
10 INPUT X$
14 GOSUB 50
16 LET R=S
20 INPUT X$
22 GOSUB 50
24 LET T=INT (S-R)
26 CLS
30 LET D=23
32 PRINT "PHYSICAL: ";
34 GOSUB 60
36 LET D=28
38 PRINT ,,"EMOTIONAL: ";
40 GOSUB 60
42 LET D=33
44 PRINT ,,"INTELLECTUAL: ";
46 GOSUB 60
48 GOTO 10
50 LET A=VAL X$(7 TO )
52 LET B=VAL X$(4 TO 5)
54 LET C=VAL X$( TO 2)
56 LET S=(A*365.25)+(B*30.6)+C
58 RETURN
60 LET P=INT ((D-(T/D-INT (T/D))*D)+.01)
64 IF P>D/2 THEN LET P=P-D/2
66 IF P=D/2 THEN LET P=0
68 PRINT "IN ";P;" DAYS"
70 RETURN
```

The biorhythm calculator for the unexpanded ZX81.

```
100 CLS
200 PRINT "ENTER BIRTHDATE"
210 PRINT ,,"IN THE FORM DD MMM YYYY"
220 PRINT ,,"I.E. 05 SEP 1942"
230 INPUT X$
235 IF LEN X$<>11 THEN GOTO 230
240 GOSUB 900
280 LET R=S
290 LET Y$=X$
300 CLS
330 PRINT "ENTER DATE OF INTEREST"
335 PRINT ,,"IN THE FORM DD MMM YYYY"
340 INPUT X$
345 IF LEN X$<>11 THEN GOTO 340
350 GOSUB 900
435 LET T=INT (S-R)
440 IF T<0 THEN GOTO 1000
445 CLS
450 PRINT ,,"BIORHYTHM  STATUS ON ";X$
455 PRINT ,,"FOR A FELLOW BORN ON ";Y$
460 LET D=23
490 PRINT ,,,,"PHYSICAL:"
510 GOSUB 2000
530 LET D=28
540 PRINT ,,,,"EMOTIONAL:"
560 GOSUB 2000
580 LET D=33
590 PRINT ,,,,"INTELLECTUAL:"
610 GOSUB 2000
700 GOTO 4000
900 LET A=VAL X$(8 TO )
910 LET T$=X$(4 TO 6)
915 GOSUB 3000
920 LET C=VAL X$( TO 2)
930 LET S=(A*365.25)+(B*30.6)+C
940 RETURN
1000 CLS
1010 PRINT AT 10,0;"THAT DAY WAS BEFORE YOUR TIME"
1020 PRINT ,,"PRESS N/L"
1030 INPUT I$
1040 GOTO 300
2000 LET P=INT ((D-(T/D-INT (T/D))*D)+.01)
2040 IF P>D/2 THEN LET P=P-D/2
2045 IF P=0 OR P=D/2 THEN GOTO 2500
2050 PRINT ,,"CRITICAL STAGE IN ";P;" DAYS TIME"
2060 RETURN
2500 PRINT ,,"TODAY"
2510 RETURN
3010 IF T$="JAN" THEN LET B=1
3015 IF T$="FEB" THEN LET B=2
3020 IF T$="MAR" THEN LET B=3
3025 IF T$="APR" THEN LET B=4
3030 IF T$="MAY" THEN LET B=5
3035 IF T$="JUN" THEN LET B=6
3040 IF T$="JUL" THEN LET B=7
3045 IF T$="AUG" THEN LET B=8
3050 IF T$="SEP" THEN LET B=9
3055 IF T$="OCT" THEN LET B=10
3060 IF T$="NOV" THEN LET B=11
3065 IF T$="DEC" THEN LET B=12
3070 RETURN
4000 PRINT AT 20,0;"ENTER P :COPY N :NEXT DAY TO 31"
4002 PRINT AT 21,0;" C :CONTINUE"
4005 IF INKEY$="" THEN GOTO 4005
4010 IF INKEY$="P" THEN GOTO 4500
4012 IF INKEY$="N" THEN GOTO 4600
4014 IF INKEY$="C" THEN GOTO 100
4015 GOTO 4005
4030 GOTO 100
4500 PRINT AT 21,0;"********************************"
4510 PRINT AT 20,0;"                                "
4520 COPY
4530 GOTO 4000
4600 IF VAL X$( TO 2)=31 THEN GOTO 4005
4620 LET V=VAL X$( TO 2)+1
4630 LET X$( TO 2)=STR$ V
4650 GOSUB 900
4660 GOTO 435
```

The biorhythm calculator program expanded to fit into a ZX81 with a memory pack.

Leprechaun's gold

An amazing game for your 16K Spectrum, courtesy of Clyde Bish of Exeter.

This 3D maze game for the Spectrum does not have dinosaurs charging up on you but it does have a Green Goblin. He follows you about, sending an indecisive player back to the start, and giving you a pot of gold if you get to the centre!

Interesting Points

1 . No maze plan is shown — There are two reasons for this: a) I think it makes the game too easy and b) the computer does not have a clue about the shape of the maze until play begins! — it all depends on the values of RND in line 210 and the moves a player makes. The maze does, however, once set, remain constant throughout the game (strings x$ and y$ ensure this).

2 . The program does not use machine code — I feel that the speed of the Spectrum's PLOT and LIN DRAW commands are fast enough. And, by avoiding machine code, the structure of the program is more easily seen by those who like to dissect programs.

3 . The problem of 16K being about 9K! — This has had to result in a compromise:
a) The elements of the maze (1000-1710) use, for the most part, numbers and are placed at the beginning of the listing so that they are displayed quickly.
b) The parts which can operate slower are put later and use variables and 'VALs' to save bytes.
c) The UDGs are set separately and are SAVEd onto tape as bytes to be LOADed in by the main program.

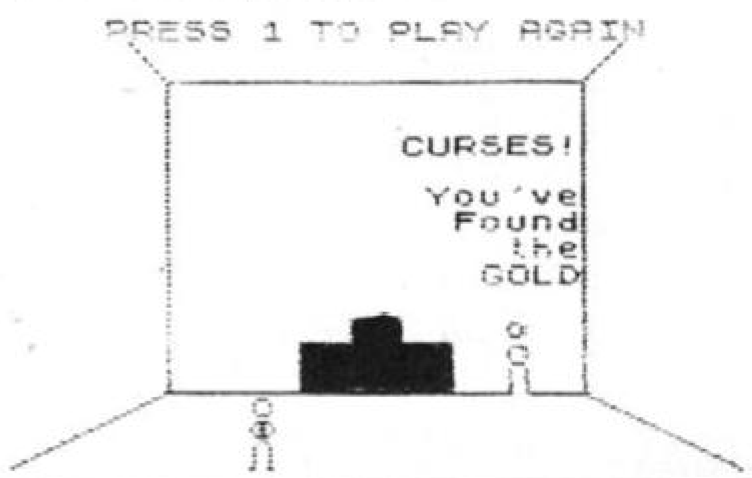
Notes on entering the program

1. As explained above the UDGs have to be set first. Type in the program starting at line 1 to line 20 and RUN this. This will set the UDGs above RAMTOP.
2. Now enter the rest of the program and SAVE it, followed by the UDG bytes using command:

SAVE "maze" LINE 9000: SAVE "maze" CODE USR"a", 168

You will get the usual 'start tape then press any key' message. Do so, but don't

This is what you should see if you manage to weave your way through the maze to the pot of gold.



Lines	Description
190-280	: Logic controlling the patch through the maze (which is set up in line 8000 — see later). The basis is this: a) 210 — If the next-but-one element is a dead-end ('7') then generate a random number. If this is less than 0.2 then the routine skips the dead-end, otherwise the dead-end is reached and the player is sent back to the beginning. b) 270 When the player reaches this decision point again, if he or she chooses to take an alternate pathway (the original choice being held in string y$) then the dead end is avoided. c) 260 If during a further attempt the player take a turning different to the first attempt (excepting (b) above) then a dead end is reached and he or she is sent back to the beginning.
1000-1710	: The elements of the maze are displayed on the screen in 3D as if looking down the passage. There are three passage elements (1000 : left turn, 1100 : right turn and 1200 : straight on) and four junction elements (1300 : forward or right, 1400 : forward or left, 1500: left or right and 1600 : forward, left or right) plus the dead-end at 1700. These are accessed in passage-junction pairs by line 200. Each element (except those at lines 1200 and 1700) are two part giving first a distant view followed by one with the player (see 4001) in a position to turn.
2000-2020	: If the player tries to walk into the passage wall this routine prevents him or her from doing so.
3000-3060	: This section invites a choice to be made (allowing only '5', '7' or '8'). If one is not made in about three seconds then the Green Goblin appears and sends the player back to the start.
4001	: This displays the player as a stick man with his hands on his hips.
4600	: A complex logic routine which, on the command '5' or '8', causes the figure to walk to the left or right and disappear around the corner of the passage.
4800	This causes the figure to move forwards on the command '7'.
5000	: This turns the figure to the left or right before walking (or bumping into the wall!).
6000-6015	: On LOADing, this is where the program begins. Instructions are given, variables are set, and the difficulty level (the number of elements in the maze) is chosen.
7000-7999	: The winning sequence. The gold is reached (much to the chagrin of the Green Goblin) and the player is invited to play again. It should be noted that the program can be stopped at any junction by pressing '0' in place of '5', '7' or '8' — see 3050.
8000	: Here the maze is set up as a string of numbers (representing the elements) held in x$. They are alternately 0-2 and 3-7, there not being two 7s following (or a junction would have two dead-end exits!).
9000	: This LOADs the user defined graphics (UDGs).

Table 1. A breakdown of the program, Leprechaun's gold, line by line.

walk away to make a cup of coffee! After about 45 seconds, the message will appear again. Don't stop the tape, just press any key and the UDG bytes will be saved. To verify use the command:

VERIFY ""

After (hopefully) the 'OK' message appears stop the tape promptly, then verify the UDGs with :

VERIFY "" CODE.

3. To LOAD simply enter LOAD "maze". The main program will LOAD, then RUN itself from line 9000 which will LOAD in the UDGs (following on tape) before displaying the instructions, and away you go.

Hope you find the GOLD!

As a final note, the single or pairs of capital letters shown in the LISTing within quotes are the UDGs and should be entered in the G mode.

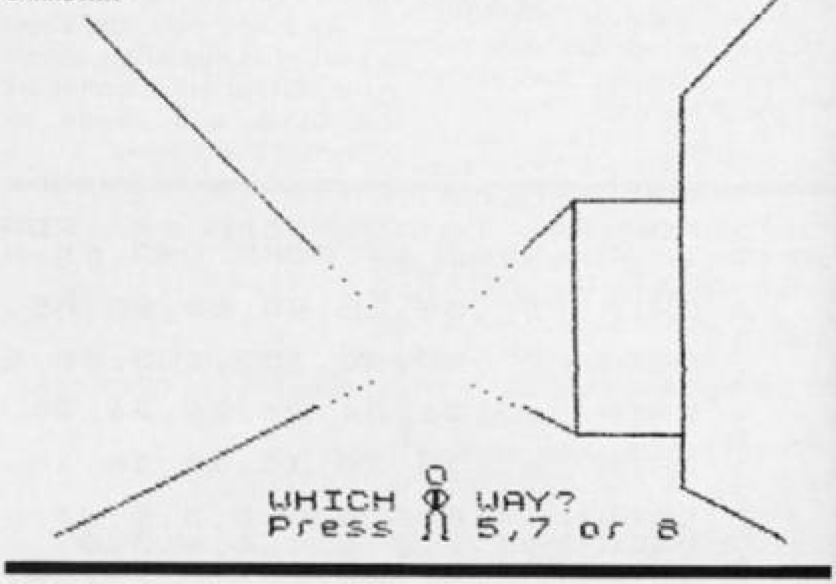
```
   1 FOR i=1 TO 19: READ p$: FOR
 n=0 TO 7: READ a: POKE USR p$+n
,a: NEXT n: NEXT i
   2 DATA "b",66,66,66,66,66,66,
66,195
   3 DATA "c",60,90,153,153,90,6
0,24,36
   4 DATA "d",24,24,24,24,24,24,
24,24
   5 DATA "e",16,16,16,16,16,16,
16,24
   6 DATA "f",8,8,8,8,8,8,8,24
   7 DATA "g",0,3,4,4,4,4,3,0
   8 DATA "h",0,192,32,32,32,32,
192,0
   9 DATA "i",3,5,9,17,33,65,1,2
  10 DATA "j",192,160,144,136,13
2,130,128,64
  11 DATA "k",2,4,4,8,8,16,16,24
  12 DATA "l",64,32,32,32,16,16,
16,24
  13 DATA "m",2,4,4,4,8,8,8,24
  14 DATA "n",64,32,32,16,16,8,8
,24
  15 DATA "o",124,124,124,124,12
4,56,68,68
  16 DATA "p",68,68,68,68,198,0,
0,0
  17 DATA "q",0,0,0,0,0,0,24,36
  18 DATA "r",36,24,60,60,60,24,
36,36
  19 DATA "s",36,36,0,0,0,0,0,0
  20 DATA "t",16,40,16,56,56,56,
40,40
 190 LET y$="
"
 200 FOR i=s TO y: GO SUB VAL x$
(i)*VAL "100"+VAL "1000": IF n>b
*v THEN GO TO VAL "200"
 205 LET i=i+s: GO SUB VAL x$(i)
*VAL "100"+VAL "1000": IF x$(i)=
"7" OR n>b*v THEN GO TO VAL "200
"
 210 IF y$(i)=" " AND x$(i+b)="7
" THEN LET y$(i)=b$: LET r=RND:
LET x$(i+b)=("8" AND r<VAL ".3")
+("7" AND r>=VAL ".3"): LET i=i+
(b AND r<VAL ".3"): GO TO VAL "2
80"
 220 IF y$(i)=" " AND x$(i+b)<>"
7" THEN LET y$(i)=b$: GO TO VAL
"280"
 230 IF y$(i)<>" " AND x$(i+b)="
8" AND y$(i)=b$ THEN LET i=i+b:
GO TO VAL "280"
 260 IF y$(i)<>" " AND y$(i)<>b$
 AND x$(i+b)<>"7" THEN GO SUB IN
T (RND*INT PI)*VAL "100"+VAL "10
00": GO SUB VAL "1700": GO TO VA
L "200"
 270 IF y$(i)<>" " AND y$(i)<>b$
 AND x$(i+b)="7" THEN LET y$(i)=
b$: LET i=i+b: LET x$(i)="8"
 280 NEXT i: GO TO 7000
1001 CLS : PLOT VAL "47",m+m: DR
AW VAL "128",a: PLOT VAL "47",VA
L "103": DRAW VAL "128",a: PLOT
x,a: DRAW m+m,m: DRAW a,VAL "119
": DRAW -u,u: PLOT VAL "240",a:
DRAW -o,m+m: DRAW a,VAL "71": DR
AW q,q
1010 LET p=w-s: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1015 IF b$<>"7" THEN LET p=w: GO
 SUB d: GO TO VAL "1000"
1020 LET c=s: LET p=m+b: GO SUB
h
1051 CLS : PLOT m,m+m: DRAW VAL
"159",a: PLOT m,VAL "103": DRAW
VAL "159",a: PLOT x,a: DRAW a,VA
L "167": DRAW -l,l: PLOT VAL "24
0",a: DRAW -o,m+m: DRAW a,VAL "7
1": DRAW q,q
1060 LET p=m+b: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1065 IF b$<>"5" THEN LET p=w-s:
GO SUB d: GO TO VAL "1050"
1070 GO SUB g: PAUSE j: RETURN
1101 CLS : PLOT v-s,m+m: DRAW VA
L "128",a: PLOT v-s,VAL "103": D
RAW VAL "128",a: PLOT x,a: DRAW
o,m+m: DRAW a,q-s: DRAW -q,q: PL
OT VAL "240",a: DRAW -VAL "32",m
: DRAW a,VAL "119": DRAW u,u
```



```
1110 LET p=w-s: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1115 IF b$<>"7" THEN LET p=w: GO
 SUB d: GO TO 1100
1120 LET c=s: LET p=m+b: GO SUB
h
1151 CLS : PLOT v,m+m: DRAW VAL
"159",a: PLOT v,VAL "103": DRAW
VAL "159",a: PLOT x,a: DRAW o,VA
L "32": DRAW a,q-s: DRAW -q,q: P
LOT VAL "240",a: DRAW a,VAL "167
": DRAW l,l
1160 GO SUB f: GO SUB e: IF n>b*
v THEN RETURN
1165 IF b$<>"8" THEN LET p=w-s:
GO SUB d: GO TO VAL "1150"
1170 GO SUB g: PAUSE j: CLS : RE
TURN
1201 CLS : PLOT 15,0: DRAW 64,32
: FOR n=2 TO 8, STEP 2: PLOT 2*n+
80,n+32: NEXT n: PLOT 239,0: DRA
W -64,32: FOR n=2 TO 8 STEP 2: P
LOT -2*n+175,n+32: NEXT n: PLOT
8,175: DRAW 72,-72: FOR n=4 TO 1
6 STEP 4: PLOT 80+n,103-n: NEXT
n: PLOT 248,175: DRAW -72,-72: :
FOR n=4 TO 16 STEP 4: PLOT 175-n
,103-n: NEXT n
```

```
1205 PRINT AT l+k,l+k;"CONTINUE"
;AT x-s,l+k;"FORWARDS"
1210 LET p=w-s: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1215 IF b$<>"7" THEN GO SUB d: G
O TO VAL "1201"
1230 LET p=m+b: LET c=a: GO SUB
h: RETURN
1301 CLS : PLOT 240,0: DRAW -32,
16: DRAW 0,119: DRAW 40,40: PLOT
 207,32: DRAW -32,0: DRAW 0,71:
DRAW 32,0: PLOT 175,32: DRAW -16
,8: FOR n=2 TO 8 STEP 2: PLOT -2
*n+159,n+39: NEXT n: PLOT 175,10
3: DRAW -16,-16: FOR n=4 TO 16 S
TEP 4: PLOT 159-n,87-n: NEXT n:
IF x$(i)="6" THEN GO TO 1402
1310 PLOT 15,0: DRAW 80,40: FOR
n=2 TO 8 STEP 2: PLOT 2*n+96,n+4
0: NEXT n: PLOT 8,175: DRAW 88,-
88: FOR n=4 TO 16 STEP 4: PLOT n
+96,87-n: NEXT n
1320 LET p=w-s: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1325 IF b$<>"7" THEN LET p=w: GO
 SUB d: GO TO VAL "1300"
1330 LET c=s: LET p=m+b: GO SUB
h
1350 CLS : PLOT 240,0: DRAW 0,16
7: DRAW 8,8: PLOT 239,32: DRAW -
64,0: DRAW 0,71: DRAW 64,0: PLOT
 175,32: DRAW -16,8: FOR n=2 TO
6 STEP 2: PLOT -2*n+159,n+40: NE
XT n: PLOT 175,103: DRAW -16,-16
: FOR n=4 TO 16 STEP 4: PLOT 159
-n,87-n: NEXT n: IF x$(i)="6" TH
EN GO TO 1451
1360 PLOT 15,0: DRAW 80,40: FOR
n=2 TO 8 STEP 2: PLOT 2*n+96,n+4
0: NEXT n: PLOT 8,175: DRAW 88,-
88: FOR n=4 TO 16 STEP 4: PLOT 9
6+n,87-n: NEXT n
1370 LET p=w-b: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1372 IF b$<>"8" AND b$<>"7" THEN
 GO SUB d: GO TO VAL "1350"
1375 IF b$<>"7" THEN GO SUB g: P
AUSE j
1378 IF b$<>"8" THEN LET p=m+s:
LET c=a: GO SUB h
1380 RETURN
1401 CLS
1402 PLOT 15,0: DRAW 32,16: DRAW
 0,119: DRAW -40,40: PLOT 48,32:
 DRAW 32,0: DRAW 0,71: DRAW -32,
0: PLOT 80,32: DRAW 16,6: FOR n=
2 TO 8 STEP 2: PLOT 2*n+96,n+40:
 NEXT n: PLOT 80,103: DRAW 16,-1
6: FOR n=4 TO 16 STEP 4: PLOT n+
96,87-n: NEXT n: IF x$(i)="6" TH
EN GO TO 1610
1410 PLOT 240,0: DRAW -80,40: FO
R n=2 TO 8 STEP 2: PLOT -2*n+159
,n+40: NEXT n: PLOT 248,175: DRA
W -88,-88: FOR n=4 TO 16 STEP 4:
 PLOT 159-n,87-n: NEXT n
1420 LET p=w-s: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1425 IF b$<>"7" THEN LET p=w: GO
 SUB d: GO TO VAL "1400"
1430 LET c=s: LET p=w-b: GO SUB
h
1450 CLS
1451 PLOT 15,0: DRAW 0,167: DRAW
 -8,8: PLOT 16,32: DRAW 64,0: DR
AW 0,71: DRAW -64,0: PLOT 80,32:
 DRAW 16,8: FOR n=2 TO 8 STEP 2:
 PLOT 2*n+96,n+40: NEXT n: PLOT
80,103: DRAW 16,-16: FOR n=4 TO
16 STEP 4: PLOT 96+n,87-n: NEXT
n: IF x$(i)="6" THEN GO TO 1660
1460 PLOT 239,0: DRAW -80,40: FO
R n=2 TO 8 STEP 2: PLOT -2*n+159
,n+40: NEXT n: PLOT 247,175: DRA
W -88,-88: FOR n=4 TO 16 STEP 4:
 PLOT 159-n,87-n: NEXT n
1470 GO SUB f: GO SUB e: IF n>b*
v THEN RETURN
1472 IF b$<>"5" AND b$<>"7" THEN
 LET p=w-s: GO SUB d: GO TO VAL
"1450"
```

Which way do you go — over to the right or ahead into the unknown?

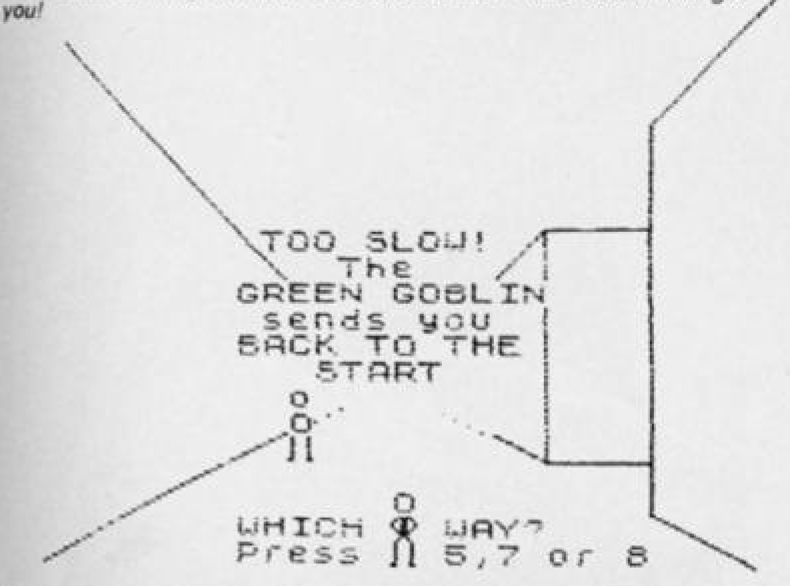



```
1475 IF b$<>"7" THEN GO SUB g: P
AUSE j
1478 IF b$<>"5" THEN LET c=a: LE
T p=m+s: GO SUB h
1480 RETURN
1501 CLS : PLOT u+l,m+m: DRAW k*
u,a: PLOT u+l,VAL "103": DRAW k*
u,a: PLOT x,a: DRAW m+m,m: DRAW
a,VAL "119": DRAW -u,u: PLOT VAL
 "240",a: DRAW -VAL "32",m: DRAW
 a,VAL "119": DRAW u,u
1510 LET p=w-s: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1515 IF b$<>"7" THEN LET p=w: GO
 SUB d: GO TO VAL "1500"
1520 LET c=s: LET p=w-b: GO SUB
h
1551 CLS : PLOT m,m+m: DRAW VAL
"224",a: PLOT m,VAL "135": DRAW
VAL "224",a: PLOT x,a: DRAW a,VA
L "167": DRAW -l,l: PLOT VAL "24
0",a: DRAW a,VAL "167": DRAW l,l
1560 GO SUB f: GO SUB e: IF n>b*
v THEN RETURN
1565 IF b$<>"8" AND b$<>"5" THEN
 LET p=w-s: GO SUB d: GO TO VAL
"1550"
1570 GO SUB g: PAUSE j: RETURN
1601 GO TO VAL "1300"
1610 LET p=w-s: GO SUB f: GO SUB
 e: IF n>b*v THEN RETURN
1615 IF b$<>"7" THEN LET p=w: GO
 SUB d: GO TO VAL "1600"
1620 LET c=s: LET p=w-b: GO SUB
h
1650 GO TO VAL "1350"
1660 GO SUB f: GO SUB e: IF n>b*
v THEN RETURN
1665 IF b$<>"7" AND b$<>"8" AND
b$<>"5" THEN GO TO VAL "1650"
1670 IF b$="7" THEN LET p=m+s: G
O SUB h: RETURN
1680 IF b$="8" OR b$="5" THEN GO
 SUB g: PAUSE j: RETURN
1701 CLS : PLOT o,w+k: DRAW VAL
"127",a: PLOT o,VAL "119": DRAW
VAL "127",a: PLOT x,a: DRAW u+l,
w+k: DRAW a,v+x: DRAW -VAL "48",
u+l: PLOT VAL "240",a: DRAW -VAL
 "48",w+k: DRAW a,VAL "95": DRAW
 u+l,u+l: IF i>y THEN RETURN
1710 GO SUB f: PRINT AT l+b,l+t;
```

```
"DEAD END";AT m-b,x-k;"Back to t
he";AT m+s,x-s;"START": PAUSE b*
v: CLS : RETURN
2000 IF b$<>"7" THEN GO SUB VAL
"5000"
2001 CLS : PLOT a,m+m: DRAW VAL
"255",a: PLOT a,VAL "135": DRAW
VAL "255",a: PRINT AT l+b,l+t;"P
ASSAGE";AT m-k,m-b;"WALL"
2005 LET p=w-b: GO SUB f
2008 IF b$="7" THEN GO TO VAL "2
020"
2010 PRINT AT w,x-k;"TURN";AT w,
w-b;"BACK";AT w+s,l+t;"Press ";(
"8" AND b$="5")+("5" AND b$="8")
: PAUSE a: IF INKEY$<>"5" AND b$
="8" OR INKEY$<>"8" AND b$="5" T
HEN CLS : GO TO VAL "2001"
2015 PRINT AT w,x-k;"      ";AT w,
w-b;"      ";AT w+s,l+t;"        ":
 PAUSE b*x: PRINT AT w-s,m;"D";A
T w,m;("F" AND b$="8")+("E" AND
b$="5"): PAUSE b*x: CLS : RETURN

2020 PRINT AT w+s,l+t;"TRY AGAIN
": PAUSE b*x: CLS : RETURN
3000 PRINT AT w,b*t;"WHICH";AT w
,w-b;"WAY?";AT w+s,b*t;"Press";A
T w+s,w-b;"5,7 or 8": FOR n=s TO
 b*v: IF INKEY$<>"" THEN GO TO V
AL "3050"
3010 NEXT n
3020 PRINT AT x,l+k; INK k;"o";A
T m,l+k;"O";AT m+s,l+k;"B": PAUS
E b*x: PRINT AT k+l,x-k;"TOO SLO
W!";AT l+b,m-b;"The";AT x-k,l+b;
 INK k;"GREEN GOBLIN";AT m-k,m-t
; INK a;"sends you";AT x-b,l+b;"
BACK TO THE";AT m-b,l+k;" START
": PAUSE v+w: RETURN
3050 LET b$=INKEY$: IF b$="0" TH
EN STOP
```

Don't take too long to make a decision or the Green Goblin will get you!




```
3055 IF b$<>"7" AND b$<>"8" AND
b$<>"5" THEN GO TO 3000
3060 PRINT AT w,l+b;"      ";AT w
,w-b;"      ";AT w+s,l+b;"      ";A
T w+s,w-b;"          ": PAUSE w: R
ETURN
4001 PRINT AT p,m;"O";AT p+s,m;"
C";AT p+b,m;"B": RETURN
4600 LET p=w-s: GO SUB VAL "5000
": FOR n=(m AND b$="8")+(x AND b
$="5") TO (w+l AND b$="8")+(INT
PI AND b$="5") STEP (b$="8")-(b$
="5"): PRINT AT w-b,n;"GH";AT w-
s,n;"IJ";AT w,n;("KL" AND b$="8"
)+("MN" AND b$="5"): PAUSE l: PR
INT AT w-b,n;(" O" AND b$="8")+(
"O " AND b$="5");AT w-s,n;(" D"
AND b$="8")+("D " AND b$="5");AT
 w,n;(" E" AND b$="8")+("F " AND
 b$="5"): PAUSE t: NEXT n: PRINT
 AT w-b,n;("G" AND b$="8")+("H "
 AND b$="5");AT w-s,n;("I" AND b
$="8")+("J " AND b$="5");AT w,n;
("K" AND b$="8")+("N " AND b$="5
"): PAUSE l: PRINT AT w-b,n;" ";
AT w-s,n;" ";AT w,n;" ": RETURN
4800 PRINT AT p,m;"o";AT p+s,m;"
O";AT p+b,m;"P";AT p+INT PI,m;"
": PAUSE w: PRINT AT p-s,m;"Q";A
T p,m;"R";AT p+s,m;"S";AT p+b,m;
" ": PAUSE w
4810 IF c=s THEN LET c=a: RETURN

4820 PRINT AT p-s,m;"T";AT p,m;"
";AT p+s,m;" ": PAUSE w: LET c=
a: RETURN
5000 PRINT AT p,m;"D";AT p+s,m;(
"E" AND b$="8")+("F" AND b$="5")
: PAUSE b*x: RETURN
6000 BORDER 4: PRINT AT 1,5; INK
 4;"[redacted]";AT 2,
5;"[redacted]";AT 3,5
;"[redacted]";AT 2,6;
 INK 0; PAPER 6;" LEPRECHAUN'S G
OLD ";AT 5,0; PAPER 7;"Can you m
ake you way to the      centre, a
nd find the GOLD?";AT 8,5;"Press
 5 to turn left";AT 9,11;"7 to m
ove forwards";AT 10,11;"8 to tur
n right";AT 11,10;"(0 to exit)";
AT 12,0;"Don't turn too early -
you'll   walk into the wall!";AT
 15,0; FLASH 1;"BEWARE!";AT 16,0
, FLASH 0;"The "; INK 4;"GREEN G
OBLIN"; INK 0;" will be         f
ollowing you. Don't take too    l
ong making a decision or he'll c
atch up and send you back to    t
he start"
6010 LET d=VAL "2000": LET e=VAL
 "3000": LET f=d+d: LET g=VAL "4
600": LET h=VAL "4800": LET j=VA
L "50": LET a=NOT d: LET s=SGN d
: LET b=s+s: LET k=b+b: LET l=k+
k: LET m=l+l: LET o=m*k: LET q=o
+l: LET t=k+s: LET u=t*l: LET v=
u+u: LET w=k*t: LET x=t*INT PI
6015 PRINT "PRESS ANY KEY TO BEG
IN": PAUSE 0: CLS : INPUT "Enter
 Difficulty Level (1 to 10)";z:
PRINT "PREPARING MAZE": LET y=z*
b+l: GO SUB 8000: GO TO 190
7000 GO SUB VAL "1700": PRINT AT
 m,x; INK VAL "6";"[redacted]";AT m+s,x-
b; INK k;"[redacted]";AT m+b,x-b;"[redacted]";AT w-s,m-t; INK a;"O";AT w
,m-t;"C";AT w+s,m-t;"B";AT 16,21
; INK 4;"o";AT 17,21;"O";AT 18,2
1;"B"
7010 PRINT AT 9,17; INK 4; FLASH
 1;"CURSES!";AT 11,18; INK 0; FL
ASH 0;"You've";AT 12,19;"Found";
AT 13,21;"the";AT 14,20; INK 6;"
GOLD"
7020 PAUSE 100: PRINT AT 1,5;"PR
ESS 1 TO PLAY AGAIN": PAUSE 0: I
F INKEY$="1" THEN CLS : GO TO 60
15
7999 STOP
8000 LET x$="
               ": FOR i=s TO y STEP
 b: LET x$(i)=STR$ INT (RND*INT P
I): LET x$(i+s)=STR$ INT (RND*k+
INT PI): IF i<INT PI THEN GO TO
VAL "8010"
8005 IF x$(i-s)<>"7" AND RND<VAL
 ".4" THEN LET x$(i+s)="7"
8010 NEXT i: RETURN
9000 LOAD "maze"CODE USR "a" : C
LS : GO TO 6000
```

Reader's reviews

A new regular feature in which you, the reader, get your chance to say what you think of the software commercially produced for your ZX micro.

This new feature has been introduced to allow you space to praise or pour scorn on any software, be it games, business, educational or domestic, that you may have tried and tested on your Sinclair ZX80, ZX81 or ZX Spectrum.

Your reviews should contain your critical thoughts about the software and the relevant details concerning the availability of the package, its price, etc. If you can provide any screen dumps to illustrate the review, so much the better. Any reviews published in this section of the magazine will be rewarded with the price of the tape you review. So, if you buy a cassette and send in a review that gets published, you'll get your software for free!

The Tomb of Dracula Moviedrome Video Maria Savage

My first complaint about this tape was that I found it difficult to load. I've had few problems in this department before, but as the program is recorded on both sides of the tape I did manage to get the program operating eventually.

The idea of the game is to find a secret treasure at the bottom of a staircase. A map is given at the beginning (the only time you are shown the map) and you find yourself in a tomb surrounded by vaults which contain pits of primeval slime, ghouls and zombies. Thirty moves are allowed before Dracula himself comes on the scene.

You accumulate, if you're lucky, silver stakes which can help you ward off the attacks of the ghouls, zombies and even Dracula! Staircases can lead to either treasure or yet another level *ad infinitum* — if you stay alive that long. Unlike other adventures, there is no choice in encounters with nasties — if you haven't enough silver stakes to defeat them, too bad.

The graphics vary from good to bad. Whilst understanding the need for an overall impression of the 300 vaults available, it would have been more enjoyable to have had greater detail on those immediately surrounding the player. However, the graphics are used well to give an impression of movement and good fight scenes with the various creatures you come across.

The game comes with a fairly detailed sheet of instructions, and it does have a certain element of suspense, eg entering a new level and no map! Help — where are the zombies?

The cassette is keenly priced at £3.95, delivery within 18 days. Finally, I must admit to finding the treasure very quickly — just luck or have they made it too easy?

Flight Simulation Psion Malcolm Jay

What does it cost to fly an aeroplane for pleasure nowadays — at least £20-30 for an hour? Well out of range of my pocket I'm afraid. So all I have are the memories of the many hours I spent cavorting around the sky some years ago. Until, that is, I got the chance to have a go on Psion's Flight Simulation, which I eagerly loaded into my 48K Spectrum. After a brief study of the instructions, I set myself to cavorting around the sky once more.

At the start of the program, you are offered the option of take off, in flight or final approach mode, and then, if you require, wind effect. The graphics are superb on this program, especially your view through the cockpit window of the horizon and the landmarks as they appear in range. The instrument panel consists of an instrument landing system, radio altimeter, landing gear indicator, flap angle, air speed, altimeter, rate of descent and climb meter, fuel guage, power setting, and finally, in the centre of the panel, a radio direction finder which is just like the real thing.

The keys on the Spectrum controlling the functions are many and various: the '5' key allows you to bank left, '6' to pitch up, '7' to pitch down, '8' to bank right, 'P' to power on, '0' to power off, 'G' to lift and lower the landing gear, 'F' to put the flaps up and 'D' to put the flaps down, 'Z' and 'X' to control the rudders, and finally, the 'M' key is used to provide you with a map of the surrounding landscape, showing the position of beacons, lakes and two runways of different lengths.

The flying controls are reasonably sensitive, although I feel that the pitch control could have been more responsive, and the left rudder control on my copy seemed extremely slow.

The object of the program is to take off, obtain and hold a reasonable altitude, fly around



the countryside or head for the other airstrip and successfully land again; I'm sure I don't need to tell you that landing is the hardest part! When one does successfully land, you are presented with the option of running the sequence again or, with re-fueled tank, taxi for take-off. This phase of the program I have yet to master — I keep getting the message 'You crashed due to taxiing too fast'.

The instructions are supplied with the package are limited to what can be fitted on the fly sheet of the cassette. Although it explains the various function keys, it is not much use to the *ab initio* pilot so I would suggest that any serious, would-be Spectrum pilot should obtain a book on flying from their local library.

So far, I have spent many hours with this program. Priced at £7.95, it is far cheaper than it would cost you to hire an aeroplane, and much safer too!

ZX81 General Statistics
ICL
Dennis Trebble

This program is available from WH Smiths at the price of £6.95. For this kind of money you would be right to expect a program of very high quality which met most of the demands imposed upon it without any problem. In fact, the program menu gives an idea of the capabilities — listing ten different statistical techniques! To be fair, one technique often superimposes into another, and the division between them is rather arbitrary, ie the scaler graph phases into both correlation and regression techniques.

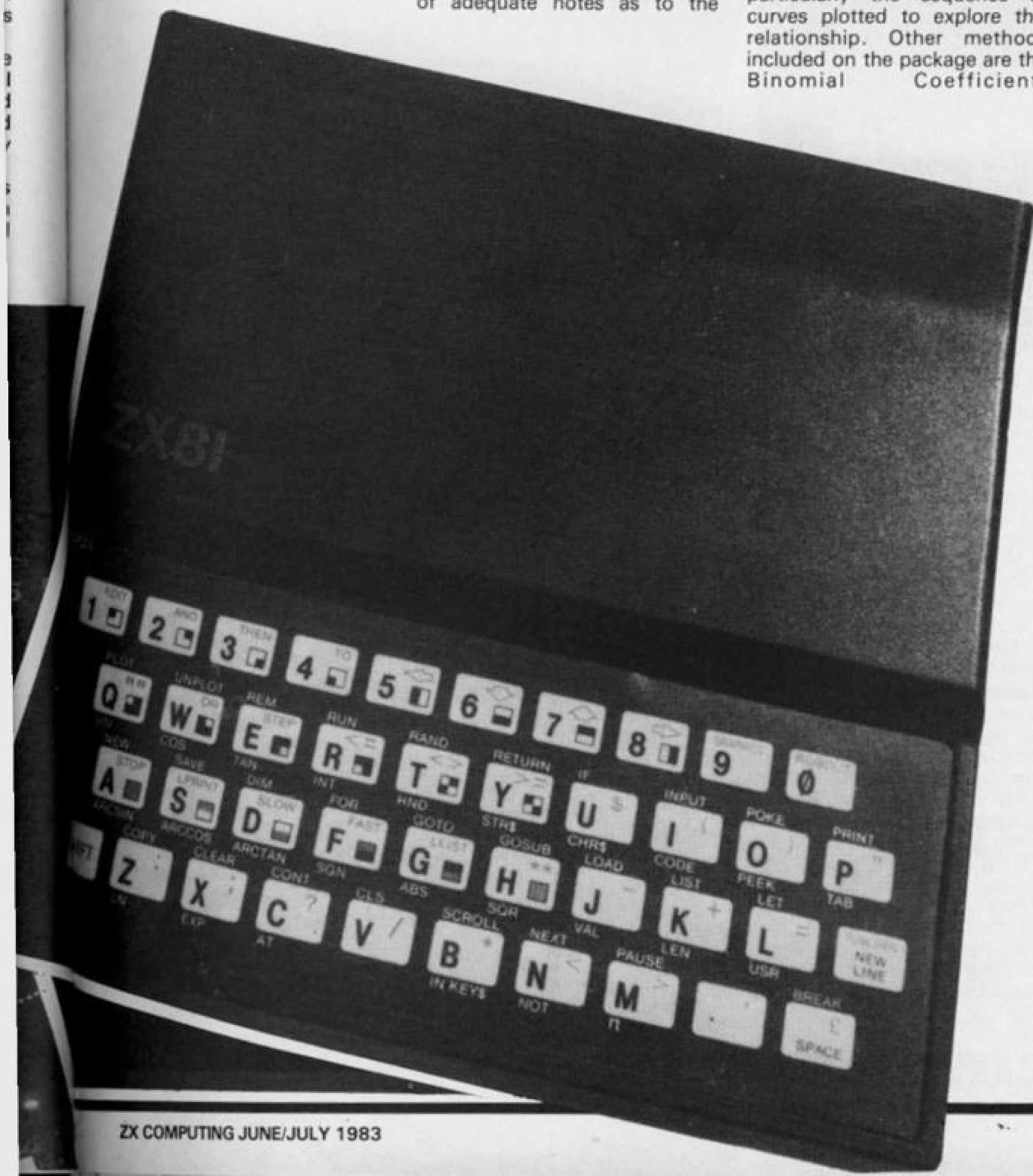
The tape is supplied with a set of adequate notes as to the running and application of the data, although it does presuppose a certain amount of statistical background from the user.

I felt the graphics displays were adequate, but could have been improved on — in particular, the section dealing with bar charts and histograms could have made greater use of the computer's capabilities to produce more satisfying visual graphics. Obviously, the graph axes form part of a subroutine that is used extensively throughout and it is possible to lose detail concerning input data as the same axes in one operation might be using a scale from 0 to ten, whilst another might use a scale from 0 to 10,000!

The section on the relationship of the mean, standard deviation and the normal curve was good — particularly the sequence of curves plotted to explore this relationship. Other methods included on the package are the Binomial Coefficient, Correlation Coefficient, Chi Squared test and Poisson Distribution. All, in some measure, were adequately explained in the printed notes accompanying the tape, but were not included as part of the on-screen instructions.

The program does have its irrating side though. For example, having plotted a scatter graph, the screen is cleared almost immediately and the graph is then seen with the regression equation printed over the top. It would have been better to ask the user to clear the screen and then, instead of the regression equation, to actually let the computer plot a line through the graphed points.

In summary, the publishers have tried to squeeze too much into too little, and in the course of doing so have made a number of compromises in fitting the whole package together. Is it good value for money? I think so, although you will need to consider carefully the amount of use you might give to the programs and reflect that a good calculator could manage the same tasks.



Hungry Horace and Gulpman
Psion and Cambell Systems
Kevin Bell

Now that Spectrum programs are making their long awaited appearance, there are two programs for all you 'Pacman' addicts. Hungry Horace from Psion and Gulpman from Campbell Systems. Firstly I'll look at Hungry Horace.

This is quite a departure from the usual 'Pacman' genre, in that the 'Pacman' is replaced by Horace, and the ghosts have been mysteriously transformed into park keepers. You have to guide Horace around the park, eating as many flowers as possible whilst evading the park keepers. Clever graphics play a large part in the game. Horace is a novel cartoon character with oversized feet, his eyes constantly searching for food. The park keepers rove around the park ready to evict Horace from the park if he is caught. Should this happen too many times then the game ends.

Occasionally, a park keeper may drop his lunch, consisting of cherries or strawberries which are even tastier than flowers and so are worth extra points. There are four mazes in all which repeat in sequence if you get past the fourth one. Somewhere in each maze there is an exit



which leads on to the next maze.

Instead of power pills there is a bell which, if eaten, causes the park keepers to panic. They turn white, their hair stands on end and they become edible for a short while. There is a bug in this routine, however, enabling Horace to get the points for eating the bell without actually eating it (if you see what I mean!). This is only a minor error in a brilliant program.

There is glorious colour throughout the program and good use is made of the Spectrum's somewhat limited BEEP command. The graphics are second to none which makes this one of the best programs for Uncle C's cantancerous box of tricks. Hungry Horace is destined for the dizzy heights of programming and attain the much sought after status of 'a classic'.

Although Gulpman is also a Spectrum 'Pacman', it takes an entirely different approach to the game. It goes for speed and versatility rather than mind-blowing graphics. Gulpman is a more traditional 'Pacman' in which a blob-guzzling dwarf is chased by four wicked ghosts, their minds full of evil thoughts. Power pills it seems have gone out of fashion, because in this game they are replaced by defence lasers. Instead of eating the ghosts you can now blow them into very small pieces, how nice!

Versatility cropped up earlier, and that is the watchword with this program. Ther is a choice of fifteen different mazes ranging from simple to downright impossible. There are two skill levels to adjust, the speed and the acceleration of the ghosts. Especially for lazy people like me who can't even be bothered to actually play the game, there is demo mode in which the hard working computer does all the playing on its own.

Again there is full colour, pretty good graphics and excellent use of sound. Full instructions are included in the program and the game is easily menu driven. An entertaining addition to any Spectrum library.

Hungry Horace and Gulpman are both priced at £5.95.

Mined-Out
Quicksilva
Chris Adam-Smith

Mined-Out equals bombed out and mind-blown. It's a game with a big hook to it — it's simply the most addictive game I've seen, and I'm not known for playing many computer games.

One of the latest titles from Quicksilva, your task is to cross several mine fields in order to save Bill the Worm who resides on the ninth level (not that I have got that far yet!). You are presented with a field which you know to be strewn with mines and all you have to do is to get across as fast as your little legs will carry you without treading on a mine; you only have one life so take care! When you move, you are given fair warning of a mine in your vicinity — it's up to you whether you heed the warning or not.

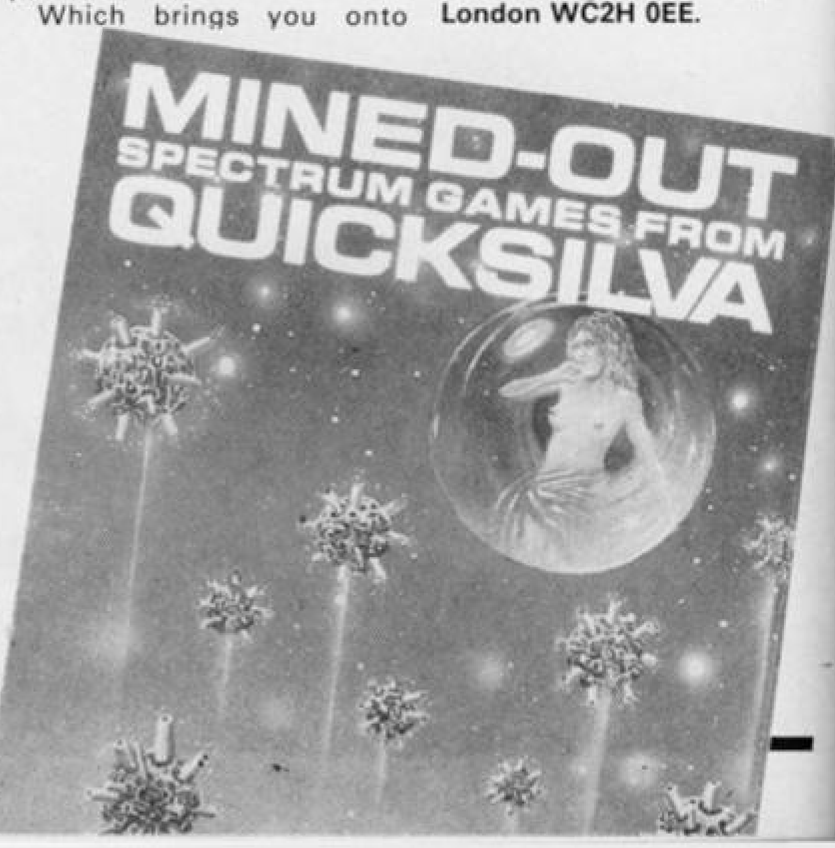
The first minefield is reasonably easy and once through you are presented with the second field, which also includes a couple of damsels in distress. These fair damsels are worth rescuing as they have bonus points in all the right places!

Which brings you onto minefield three. Oh heck! This includes all the things that have gone before as well as a mobile bomb, which although a bit erratic in its movements, pursues you with definite malice aforthought. Gentle audio noises and a flashing warning tell you that bombs are around you, but this doesn't seem to make it easier when there's a bomb on your tail. It's a bit like being at the seaside when you are running over a deep pool of water using rocks for stepping stones — you tend to speed up when dry land is in sight, and that's the rub.

I did actually make it to level four, but I didn't stay very long before...BOOM! Oh well, at least I got the option to go back to either levels one, two or three.

Apparently it gets much worse on the other levels of the game, but if I told you how much worse it might put you off having a go. All in all, a simple game and very thoughtful — a bit like Mastermind with bombs under your chair should you 'pass'. A game where you really don't need someone offering advice and breathing down your neck.

Minded-Out is priced at £4.95.



The authors of any of the reviews published within this section of the magazine will receive payment equal to the software reviewed. All contributions should be typed, double spaced, and be accompanied with screen dumps where possible. Your reviews should be sent to the following address:

Reader's reviews,
ZX Computing,
145 Charing Cross Road,
London WC2H 0EE.

Six of the best

Ms Goodridge of Rayne presents six eductional programs for your 1K ZX81.



These six programs were written for my young daughter to learn some basic mathematics while still having some fun. I have tried to use graphics wherever I could, but I'm sure you know the limitations of the 1K ZX81 as well as I do!

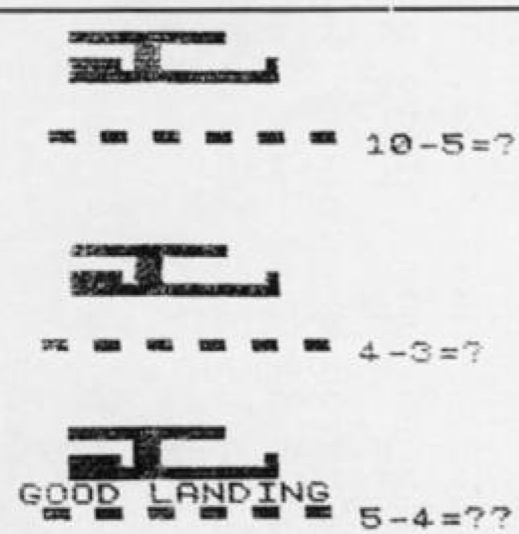
Four of the progams deal with subtraction, one with co-ordinates and the last with division. The first of the programs is called Helicopter.

Helicopter

This is a simple subtraction game with a graphic reward. When the program is RUN, a graphic 'copter appears over some ground giving the illusion of movement. The object of the game is to answer the questions correctly in order to get the helicopter to land. If three questions are answered wrongly then the message 'crash' is printed at the top of a cleared screen. If all the answers are correct then the 'copter lands and the game begins again.

The level set is easy, as is the entry of answers, which is all one touch, with no use of Newline. This is achieved by the CODE facility acting on INKEY$ inputs. Spurious inputs produce strange results as the CODE values are often quite different, but I'm afraid in 1K there is not enough room for full error trapping.

```
  1 LET A$="            "
  2 LET B$="            "
  3 LET C$="            "
  4 LET D$="            "
  5 LET E$="            "
  6 LET C=PI/PI
  7 LET D=15
  8 LET E=C
  9 LET F=0
 10 RAND
 20 LET A=(INT (RND*10)+C)
 30 LET B=(INT (RND*A)+C)
 40 PRINT AT E,C;A$
 50 PRINT AT E+C,C;B$
 60 PRINT AT E+2,C;C$
 80 IF E=11 THEN GOTO 500
150 PRINT AT D,C;E$;AT D,C;D$;A
;"-";5;"=?"
155 IF NOT INKEY$="" THEN GOTO
200
160 GOTO 150
200 IF CODE INKEY$-28=A-B THEN
LET E=E+C
210 IF CODE INKEY$-28<>A-B THEN
LET F=F+C
215 IF F=3 THEN GOTO 400
220 GOTO 20
400 CLS
410 PRINT "CRASH"
420 STOP
500 PRINT "GOOD LANDING"
510 PAUSE 150
520 CLS
530 RUN
```



Each correct answer you give in the program, Helicopter, the closer you bring the 'copter to a safe landing.



Ants

The third program, Ants, begins by displaying the words 'There are "x" ants', x being the number selected by the computer up to a limit of 20. The ants are then displayed, and if x was nine then nine ants are shown. Immediately the ants are all on the screen, the computer selectes a number smaller than x and blanks out that many ants. The word 'TAKEN?' then appears at the bottom of the screen, and the number of ants removed has to be input using Newline.

If this is correct the question 'LEFT' appears, to which the answer is the number of ants left on screen. If this is answered correctly then the game begins again. In the case of incorrect input, the word 'WRONG' appears and the game begins again.



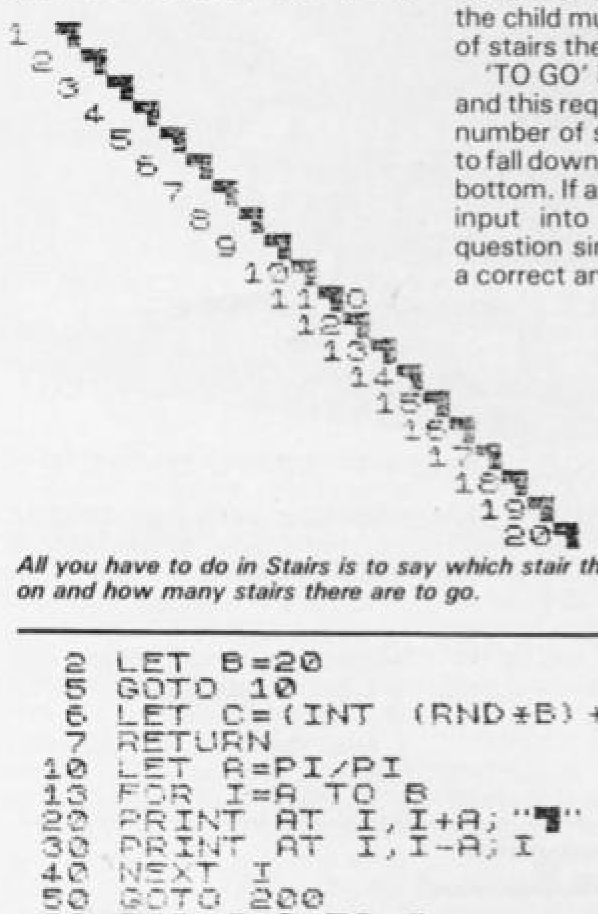
There were 12 ants on-screen to begin with – how many are now left?

```
   1 LET B$="■■"
   4 LET D=PI/PI
   7 GOTO 100
  10 LET A=(INT (RND*20) +D)
  20 RETURN
  30 LET E=D
  32 LET F=D
  35 FOR I=D TO B
  40 PRINT AT E,F;B$
  55 IF I=D THEN GOTO 65
  60 IF INT (I/3) =I/3 THEN LET E
=E+3
  65 LET F=F+6
  70 IF F>=18 THEN LET F=D
  80 NEXT I
  85 RETURN
 100 GOSUB 10
 105 IF A<=4 THEN GOTO 100
 110 LET B=A
 120 GOSUB 10
 130 IF A>=B-D THEN GOTO 120
 135 LET C=A
 140 CLS
 150 PRINT "THERE ARE ";B;" ANTS
"
 155 GOSUB 30
 156 LET G=B
 160 LET B=C
 170 LET B$="     "
 190 GOSUB 30
 200 PRINT AT 20,D;"TAKEN?"
 210 INPUT Q
 220 IF Q=B THEN GOTO 260
 230 PRINT AT 21,D;"WRONG"
 240 PAUSE 100
 250 RUN D
 260 PRINT " LEFT?"
 270 INPUT R
 280 IF R=G-B THEN RUN D
 290 GOTO 230
```

Stairs

In Stairs, a flight of stairs appears, with the numbers '1 to 20' next to them. (This number can be increased, but would still have to be in the range of 20, say from 60 to 40.) A ball then bounces down the stairs, and comes to rest at a certain point. 'DOWN' comes up at the bottom of the screen, to which the child must input the number of stairs the ball descended.

'TO GO' is the next question, and this requires the input of the number of stairs left for the ball to fall down before it reaches the bottom. If an incorrect answer is input into this program, the question simply reappears until a correct answer is entered.



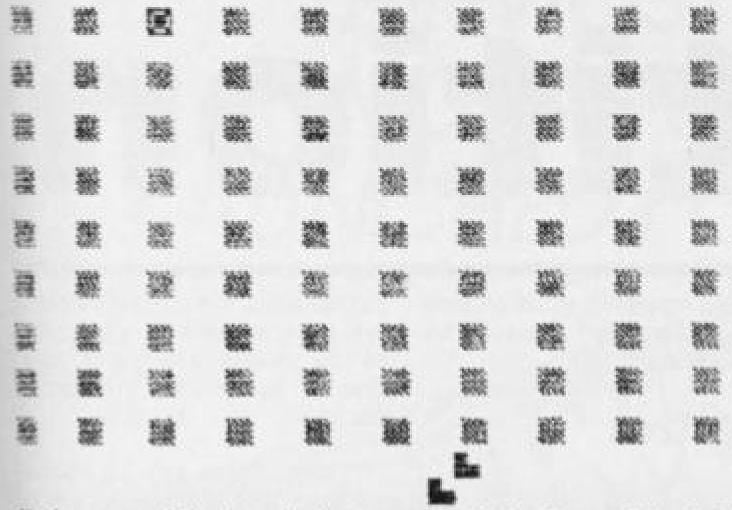
All you have to do in Stairs is to say which stair the ball has stopped on and how many stairs there are to go.

```
   2 LET B=20
   5 GOTO 10
   6 LET C=(INT (RND*B) +A)
   7 RETURN
  10 LET A=PI/PI
  13 FOR I=A TO B
  20 PRINT AT I,I+A;"■"
  30 PRINT AT I,I-A;I
  40 NEXT I
  50 GOTO 200
  60 FOR I=A TO C
  70 PRINT AT I-A,I+A;"O"
  80 PRINT AT I,I+2;"O"
  90 PRINT AT I-A,I+A;" "
 100 NEXT I
 110 RETURN
 200 GOSUB 6
 210 GOSUB 60
 220 PRINT AT B+A,A;" DOWN?"
 230 INPUT Q
 240 IF NOT Q=C THEN GOTO 230
 250 PRINT AT B+A,A;" TO GO"
 260 INPUT R
 270 IF NOT R=B-C THEN GOTO 250
 280 PRINT AT B+A,A;"GOOD    "
 300 FOR I=A-A TO B+A
 310 PRINT AT I,I+A;"   "
 320 NEXT I
 330 RUN 2
```

Fireman

Fireman is the first real 'program' I wrote, and as such I must beg forgiveness for the untidy layout of the listing. Also, there are no on-screen instructions. This is because the game uses quite a large display and memory was pretty tight when it was finished!

When RUN, a grid of grey co-ordinate squares appears, and in one of them a face appears. By INPUTting, using Newline, first the row number and then the column, the man can be rescued, and he walks off at the bottom of the screen. The face is an inverse 'O', by the way.



Having rescued the face in the top row, third column of the building, you are rewarded by seeing the legs of the figure happily walking away.

```
  1 LET T=1
 10 FOR Z=T TO 18 STEP 2
 20 FOR Y=T TO 30 STEP 3
 30 PRINT AT Z,Y;"▒"
 40 NEXT Y
 50 PRINT
 60 NEXT Z
 70 LET A=2*INT (RND*9)-T
 80 LET B=3*INT (RND*10)-2
 85 IF B<=3 THEN LET B=T
 86 IF A<=3 THEN LET A=T
 90 PRINT AT A,B;"▒"
100 INPUT R
110 INPUT C
120 IF NOT R*2-T=A AND C*3-T<>B
THEN GOTO 300
140 FOR X=15 TO 25
150 PRINT AT 19,X;" █ "
160 PRINT AT 20,X;" █ "
170 NEXT X
176 CLS
180 GOTO 10
300 PRINT "WRONG"
```

Rising numbers

Rising numbers begins with three numbers displayed on the screen, say a 20 at the top left, and next to the top left number a 0, about five PRINT spaces away. Underneath these, at line 15, another number will appear, say three. The top left number, 20, represents the original number and the number at the bottom represents 20 after a mystery number has been subtracted from it, leaving three. The object of the game is to guess what number was subtracted from 20 to leave three, and to make this harder the number at the bottom of the screen begins to rise. If it reaches the top before the correct number has been input then the message 'YOU MISSED THE NUMBER' comes up.

If the number is correctly guessed then 'YOU HAD "x" LIVES LEFT' is the line displayed. Entering numbers on this game is again by INKEY$ and this is where the zero to the right of the original number comes in. Because it is not possible to input a number greater than 9 with INKEY$, and because I wanted the display to be continuous, the first number input is altered if the value of the answer is expected to be equal to 10 or greater.

In the example given the required input would be 17. The first number put in is 1. As 17 is greater than 10, the INKEY$ 1 is multiplied by ten and displayed in place of the zero. The next number input, which is seven, is simply added to the 10 to produce 17. RUN the program to see how it works!

```
  1 REM MOVING SUBTRACTIONS
  9 LET T=0
 10 LET A=PI/PI
 15 CLS
 20 LET B=(INT (RND*20)+8)
 30 LET C=(INT (RND*B)+1)
 40 LET D=B-C
 50 FOR I=20 TO A STEP -A
 60 PRINT AT A,A;B;"    ";T
 70 PRINT AT I,A;D
 80 PRINT AT I+A,A;"  "
 85 IF INKEY$<>"" THEN GOSUB 200
100 NEXT I
110 PRINT "YOU MISSED THE NUMBE
R"
120 PAUSE 60
130 RUN 9
200 LET F=CODE INKEY$-28
210 IF T=0 AND C>=10 THEN LET F
=F*10
220 LET T=T+F
230 IF T=C THEN GOTO 300
240 RETURN
300 PRINT AT A,A;B;"    ";T
306 PRINT "WELL DONE YOU HAD ";
I;" LIVES LEFT"
310 FOR J=A TO 50
320 NEXT J
330 CLS
340 RUN 9
```

```
 16    5
WELL DONE YOU HAD 7 LIVES LEFT
```

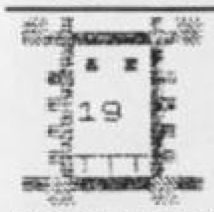
If you take five from 16 you are left with 11. Be quick about your answer or else you lose a life!

Monster division

When you play Monster division the robot appears, and in his chest is a number. The sign 'IF DIVISOR ENTER NUMBER ELSE 0' is at line 1, and this means that if the number displayed at the bottom of the screen is an exact divisor of the number in the chest of the robot, then that number should be entered using Newline. If the number is not an exact divisor, say three into 19, then the number '0' should be entered. All the numbers smaller than, and including, the number shown on the robot are tested, a lengthy process with numbers larger than 10, but good practice for little ones.

If you get all the divisors correct the message 'YOU KILLED HIM' is shown, and if you get one wrong 'HE KILLED YOU' and the game stops.

```
  1 LET A=PI/PI
  3 LET B=0
  4 LET C=4
  5 PRINT "IF DIVISOR ENTER NUM
BER ELSE 0"
 10 LET N=INT (RND*20)+1
 20 DIM F(N)
 30 FOR G=A TO N
 35 LET B=B+A
 40 IF INT (N/B)=N/G THEN LET F
(G)=G
 50 NEXT G
 60 PRINT "▒█████▒"
 65 PRINT " █ ▪ ▪ █ "
 70 FOR Z=A TO C
 80 PRINT " █   █"
 90 NEXT Z
100 PRINT "█▒████▒█"
110 PRINT AT C,2;N
115 PRINT AT 6,2;"TTT"
120 FOR Z=1 TO N
125 PRINT AT 20,2;Z;"?"
130 INPUT Y
140 IF NOT Y=F(Z) THEN GOTO 300
150 NEXT Z
160 PRINT "YOU KILLED HIM"
161 FOR O=1 TO 25
162 NEXT O
163 CLS
170 GOTO 1
300 PRINT "HE KILLED YOU"
```




```
IF DIVISOR ENTER NUMBER
ELSE 0
```

A sample screen illustration from the program, Monster division, showing the robot with the number 19 on its chest.

Competition

Your chance to win some of the latest Sinclair software for your ZX81 or ZX Spectrum.

Here's a competition that's a little out of the ordinary! And it's your chance to be a little creative with the English language. No, I don't want you to write a book or anything — simply finish off the two limericks I've started.

I'm sure that you'll have come across a limerick before but for the benefit of those who haven't a limerick is a five line verse in which the first, second and fifth lines must rhyme with each other and the third and fourth lines must rhyme with each other. For example, this is the kind of poetry we're after:

There was a programmer from Crewe,
Whose achievements were all too few,
His listings were bad,
Which made him feel sad,
So don't let that happen to you!

only much better, of course.

To enter the competition, you have to finish off the two limericks in the box below and send it off to our Charing Cross Road address. The winners will be chosen by the Deputy Editor (that's me folks) and be warned, I have a weird sense of humour!

Obviously, I would like to publish the results of the competition so it would be nice if you could keep them clean. However, should you feel otherwise inspired, your entries will be appreciated around the editorial offices.

The Prizes

There will be three prizes in this competition, and all involve your choice of up to £50.00 of Sinclair Research's range of software, the second prize winner will receive up to £30.00 worth, and the third prize winner will have a choice of up to £20.00 worth of their software.

Rules

This competition is open to all UK and Northern readers of ZX Computing except employees of Argus Specialist Publications Ltd, their printers and distributors, employees of Sinclair Research Ltd, or anyone else associated with the competition.

As long as the correct coupon is used for each entry, there is no limit to the number of entries.

All entries must be postmarked before July 31st, 1983. The prizes will be awarded to the best three entries, the decision to be made by the Editor of ZX Computing. No correspondance will be entered into with regard to the results and it is a condition of entry that the Editor's decision is accepted as final.

The winner will be notified by post and the results will be published in a future issue of ZX Computing.

Address your answers to:

ZX Competition — Limericks,
145 Charing Cross Road,
London WC2H OEE.

Results

Congratulations to the following, Bdr. AJ Evans, Mahmud Adat and Mr R Pickering, for their amazing entries to the Feb/March competition. I would have loved to have printed all the words you made out of the Spectrum Keyword 'RANDOMIZE' but I'm afraid I didn't have the space! Your prizes are winging their way to you by first class mail. Very well done.

My thanks to everyone else who entered the competition and my apologies for the fact that you all couldn't win something. And yes, I am quite embarrassed about leaving *space for 40 words, as you all* managed to get more than that!



Competition

Name . . . .

Address . . . .

. . . .

. . . . Postcode . . . .

Finish off the following two limericks:

There was a young lady from Tooting,
Who got all her kicks from computing,

. . . .

. . . .

. . . .

There was a young man from Hyde,
Who viewed his computer with pride,

. . . .

. . . .

. . . .

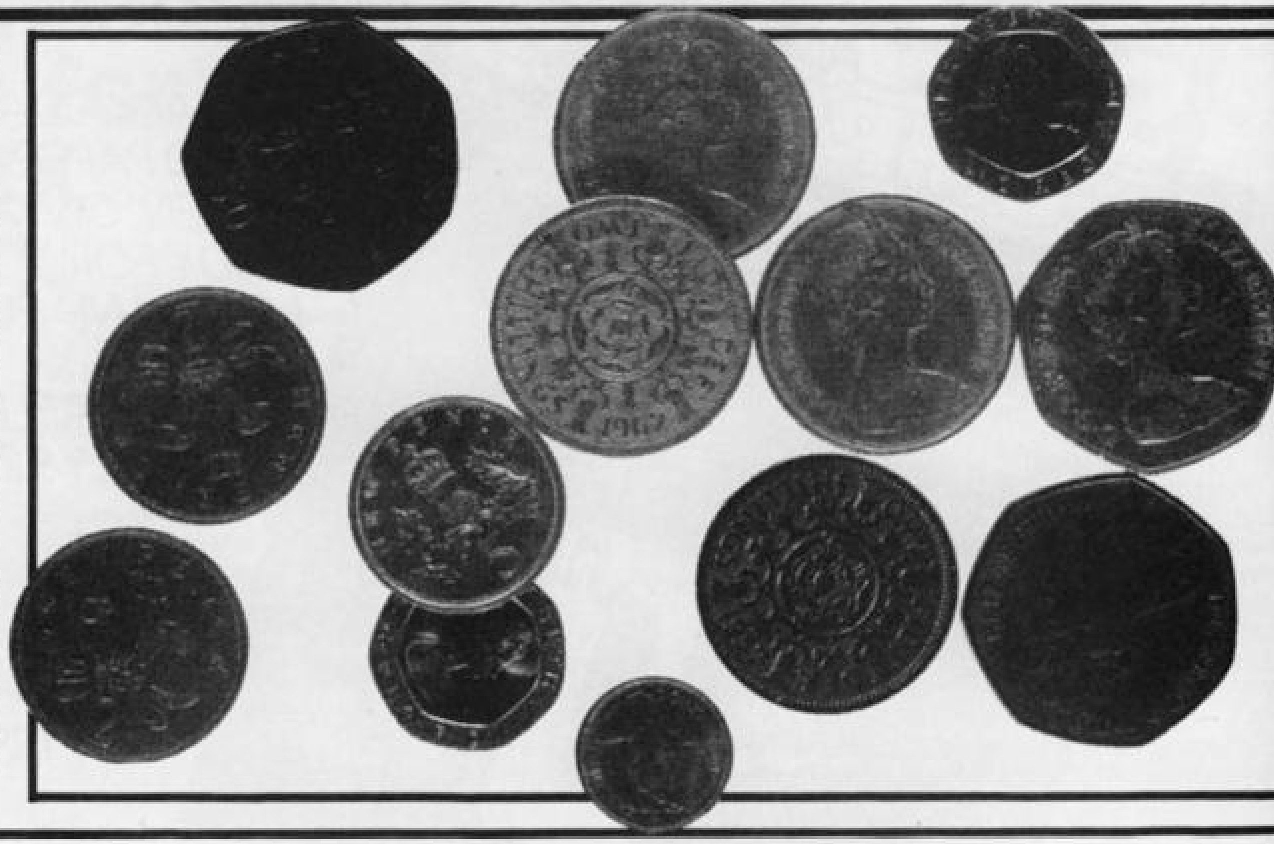
Bank statement

Keep one step ahead of your spending with this program written by Mr AP Walton of Selby.

This program, written for the ZX81, has been designed to give output with the appearance of the type of statement sent out by the banks.

The action of the program should be fairly self-explanatory from the listing below. However, here are a few simple instructions. After the program has asked for the date of a particular transaction, the question 'TYPE?' allows you to enter a cheque, credit, etc. Only the first five letters of the response will be printed, so you might like to think up some titles which will be meaningful to you.

Withindrawals should be entered as minus amounts, for example, −10.25 would represent a withdrawl of £10.25. Overdrawn amounts are printed in inverse video. The entire program is 1069 bytes long.



```
  10 LET B=0
  20 DIM T$(4)
  30 DIM D$(5)
  40 GOSUB 1000
  50 PRINT AT 0,3;"DATE?"
  60 INPUT D$
  65 REM STOP IS SHIFTED A
  70 IF D$(1)=" STOP " THEN GOTO
 2000
  75 PRINT AT 0,3;"TYPE?"
  80 INPUT T$
  90 PRINT AT 0,3;"AMOUNT?    "
 100 INPUT V
 110 SCROLL
 120 PRINT AT 0,0;"
 "
 125 REM THAT WAS 32 SPACES
 130 GOSUB 1000
 140 LET C=16+6*(V<0)
 150 LET P=ABS V
 160 PRINT AT 20,0;D$;" ";T$;
 170 GOSUB 9000
 180 LET B=B+V
 190 LET P=B
 200 LET C=31
 210 GOSUB 9000
 220 GOTO 40
1000 PRINT AT 1,0;"DATE TYPE";TA
B 13;" CR ";TAB 20;" DR ";TAB 27
;"BAL  "
1010 RETURN
2000 PRINT AT 0,0;"N/L TO SAVE "
STATEMENT""
2010 INPUT C$
2020 SAVE "STATEMENT"
2030 PRINT AT 0,0;"
 "
2035 REM THAT WAS 23 SPACES
2040 GOTO 40
9000 LET KL=INT (ABS P+.005)*SGN
 P
9010 LET KP=INT ((ABS (P-KL)*100
)+.5)
9020 LET Z$=STR$ KP
9030 LET Z$=STR$ KL+"."+("0"+Z$)
(LEN Z$ TO )
9035 IF Z$(1)="-" THEN GOTO 9060
9040 PRINT TAB (C-LEN Z$+1);Z$;
9050 RETURN
9060 LET Z$=Z$(2 TO )
9070 FOR X=1 TO LEN Z$
9080 LET Z$(X)=CHR$ (CODE Z$(X)+
128)
9090 NEXT X
9100 GOTO 9040
```

```
    DATE?
DATE TYPE        CR        DR       BAL
AUG   OPEN  123.45                123.45
18/8  CHEQ              24.99      98.46
19/8  POOL  200.00                298.46
25/8  CHEQ             300.00       1.54
26/8  PAY    80.00                 78.46

N/L TO SAVE "STATEMENT"
```

A screen illustration from the program, Bank statement.

campbell systems

THE VERY BEST IN MACHINE CODE FOR THE **ZXSPECTRUM** & **ZX81**

All programs supplied double-recorded and mailed 1st class by return. Prices include VAT and postage within Europe. SAE for full list.

for 48K Spectrum **MASTERFILE** business/domestic filing and reporting system. So flexible that it is equally usable for your mailing lists, catalogues, stock control, text extracts ...... applications are endless. Fully user-defined data and report display formats, dynamic variable-length file, records and data items. Fully menu-driven with powerful search facilities, sorting, total/average, update, multiple independent files, printing. Yes, we aim to support microdrive when Uncle delivers. Nearly all the 8K we use is machine code, so you get 32K per file. Comes with example file and 22-page manual. £15.00

NEW Masterfile-16 for 16K Spectrum owners. We have produced a subset version of MASTERFILE called MASTERFILE-16. It has most of the features of big MASTERFILE, with 7 pre-defined report formats and 6 pre-defined fields e.g. name, address, etc., and we have found a useful 4500 bytes of file space. MASTERFILE and MASTERFILE-16 are fully 'file-compatible' and will happily process each other's files, so that 16K owners who upgrade to 48K can use their old files immediately. Complete with 12-page manual, priced at £8.95

PLEASE REMEMBER TO STATE CLEARLY WHICH VERSION OF MASTERFILE YOU REQUIRE.

for 16K Spectrum **GULPMAN** game of the '.....man' variety with 15 mazes, 4 chasers, laser defence, 9 grades, 9 speeds, demo mode, choice of joystick control. "An extraordinarily good program" raves Boris Allan for Popular Computing Weekly. We think you will agree. £5.95

for 16K ZX81 **GULP 2** almost identical spec to GULPMAN. £4.75

for 16-64K ZX81 **THE FAST ONE** is the predecessor to MASTERFILE and is in use all over the world now. Specification is very similar to MASTERFILE. £12.00

Campbell Systems, Dept.(zx), 15, Rous Road, Buckhurst Hill, Essex, IG9 6BL, England. Telephone: 01-504-0589

CS

MICROSPHERE

MICROSPHERE COMPUTER SERVICES LTD
72 ROSEBERY ROAD · LONDON N10 2LA
TELEPHONE 01-883 9411

EVOLUTION (48K Spectrum)

Can you: Trace the course of evolution?
Start with primaeval soup & end up with man?
Survive the odd major disaster?
Should be easy... after all it's been done before!

Only £6.95

OMNICALC (48K Spectrum)

The best spreadsheet analysis program you can buy for the Spectrum. Uses include financial planning, home budgets etc.

Features:
- 99 columns or 250 rows
- Fully prompted input
- Extensive repeat facilities
- Conditional expressions and RND
- Totalling and sub-totalling
- Comprehensive manual

£9.95

ZX-SIDEPRINT (For Spectrum or ZX81)

Print sideways on the ZX-Printer to give proper 80/132 etc.character print lines.
- Easy to add to existing programs
- Routines to take print lines direct from screen
- Lower case letters even on a ZX81!

(State version req'd) £4.95

CREVASSE & HOTFOOT (16K Spectrum)

2 original games from Microsphere

Can you make it across the ice-cap whilst chasms open and close all around?
Help the rabbit get to fields full of carrots - but remember: the plumper the rabbit, the greater the danger.

Excellent value £4.95

THE TOMB OF DRACULA!



3D HORROR ADVENTURE GAME!

For the **16K ZX81** or with colour and sound for the 38K SPECTRUM a superb 3D graphics adveture game! **Enter** Dracula's tomb at 30 minutes to sinset . . . **wander** through the tom's pre-mapped 300 vaults in search of the fabled Vampire's Treasure . . . **pick up** valuable silver stakes and use them to defend yourself against the lurking horrors . . . ghouls, zombies, pits of primaeval slime. . . . **See** them all on the computer's plan of the tomb . . . when it will let you! **Take** a change on a Mystery Vault . . . if you dare! And all the time the minutes are ticking by to sunset . . . when **Dracula** rises from his coffin and comes after **you!** Each of the infinite levels of the tomb has its own 300 vaults . . . go as deep as you like, the Prince of Darkness will seek you out in his blood-lust! **WARNING: people of an exceptionally nervous disposition should play this game only during the hours of daylight!** Special facility lets you save a game in progress onto tape so you can continue it whenever you choose.

For ZX81 with 16K RAM.......................... £3.95
For 48K SPECTRUM.............................. £4.95

Price includes ready-to-load cassette with library case and inlay, full instructions, postage and packing. **Order today!** Money refunded if not delighted! **Send cash, P.O. or cheque** to:

FELIX SOFTWARE (Dept. ZX 5)
(incorporating MOVIEDROME VIDEO)
19 Leighton Avenue, Pinner, HA5 3BW.

DON'T MISS THIS INCREDIBLE OFFER!

50 GAMES £9·95



YES, 50 GAMES! on cassette for all the following:

GALACTIC ATTACK
SPACE MISSION
LUNAR LANDER
PLASMA BOLT
STARTREK
RADAR LANDING
ATTACKER
GALACTIC DOGFIGHT
ZION ATTACK
IVASIVE ACTION
OXO
BOGGLES
PONTOON
SKI JUMP
HANGMAN
OLD BONES
THIN ICE
MAZE EATER
CRITTER
MOTORWAY
FORCE FIELD
NIM
TUNNEL ESCAPE
BARREL JUMP
CANNONBALL BATTLE
OVERTAKE
SITTING TARGET
SMASH THE WINDOWS
SPACE SHIP
JET FLIGHT
PHASER
INTRUDER
INFERNO
GHOSTS
SUBMARINES
ROCKET LAUNCH
PLANETS
BLACK HOLE
DYNAMITE
DO YOUR SUMS
DERBY DASH
SPACE SEARCH
UNIVERSE
RATS
TANKER
PARACHUTE
JETMOBILE
HIGH RISE
THE FORCE
EXCHANGE

sinclair SPECTRUM
apple (ON DISC & CASSETTE)
ACORN ATOM
BBC
DRAGON
ATARI VIC 20
SHARP
ORIC-1
NEW BRAIN
ZX81 LYNX



POSTAGE FREE TO U.K.
ORDER DIRECT TO:
U.T.S.
P.O. Box 96
53 Lincoln Grove
Harrogate
North Yorkshire
HG3 2UZ

PLEASE SEND ME ____
CHEQUE ENCLOSED, VALUE ____
NAME ____
ADDRESS ____
POSTCODE ____ G/221

cut & post now

FOOTBALL POOLS AND HORSE RACING FORECASTING



SPECTRUM AND ZX81



These massive programs use every inch of memory to process football and horse racing data to provide you with pools forecast results to enter on your coupon, or a prediction of which horse represents the best bet. The booklets provided give step-by-step instructions to help you towards getting THE BIG WIN. Don't waste your money on games that soon get boring — invest now in these efficient and complex statistical analysis and sorting programs that you can use for years to come to give endless interest and entertainment. The author, Professor George, is lecturer in statistical studies and a leading authority on sporting statistics and forecasting. Have his valuable experience working for you.

FOOTBALL ANALYSIS for ZX81 (16K) and Spectrum (16K)
HORSE RACING ANALYSIS FOR ZX81 (16K) and Spectrum (16K)

Both versions £15 each inclusive. Outside UK plus £1.50.
Send cheque/P.O. to:
Holly Products, Blackthorn House,
Dukes Lane, Gerrards Cross, Bucks SL9 7JZ.
Allow 28 days for delivery (normally 7-10 days)
Every ZX user should have these valuable and educational programs

THE ART OF PROGRAMMING THE 1K ZX81

This book shows you how to use features of the 1K ZX81. We explain its random number generator, how to use the graphics, look at PEEK and POKE and the ZX81's timer. We also include a number of complete ready-to-run programs designed to fully exploit the capabilities of your ZX81.



Send order form with remittance to:
ASP Ltd, 513 London Road,
Thornton Heath, Surrey CR4 6AR.

Please rush me a copy of The Art of Programming the 1K ZX81. I enclose my cheque/postal order made payable to A.S.P. Ltd for £2.20 (inc 25p post and packing).

Name ....

Address ....

....

....

Please allow up to 14 days for delivery.

JOIN THE USERS' CLUB

FREE BOOK!

Make the most of your Spectrum or ZX81 by joining the country's strongest and most enthusiastic users' club—the National ZX Users' Club.

When you join us, we'll send you our 'new members welcome pack' with 30 programs (15 for the ZX81, 15 for the Spectrum), plus the first issue of our value-packed club magazine INTERFACE.

Inside INTERFACE as well as at least six programs for your ZX81 or Spectrum, you'll find letters from members as they share tips and ideas, and sound off on a variety of subjects (such as you-know-who's appalling delivery times record), a list of local ZX clubs, and special offers from software houses and book publishers—special offers just for members.
If you're having problems with your computer, we have experts on tap to try and help you. If you want to start a local users' club, we'll give you publicity, and let you buy books at a special discount for your club members.
If you're not a club member, you're missing out on making the most of your micro (and try saying that ten times quickly!) Come on in and join us, the water's fine. It's £9.50 for a year's INTERFACE, and we'll send you a sample issue for £1.00

Run by Tim Hartnell, the National ZX Users' Club is a resource just waiting to be tapped. Come on and start tapping.

National ZX Users' Club,
Dept. **ZC**
44-46 Earls Court Road,
London, W8 6EJ

OK, Tim, you've convinced me:

( ) Send me my 'new members' welcome pack' plus my first issue of INTERFACE and keep those INTERFACES coming for the next year. I enclose £9.50 (UK), £12.50 (Europe)

( ) Just send me a sample of INTERFACE. I enclose £1.00

I have ( ) a ZX81 ( ) a Spectrum

Name ....

Address ....

....

Stateside

Tim Hartnell reports from the West Coast Computer Faire in San Francisco.

The eighth West Coast Computer Faire, held at the end of March in an unusually wet San Francisco, was a desert for owners of ZX or Timex-Sinclair computers. Neither Timex nor Sinclair bothered to put in an appearance, and the only support for the machines came from two independent companies.

The New York based software company, Softsync, had a stand at the show, selling a range of ZX81/Timex-Sinclair 1000 software, most of it produced in the UK. Reston Publishing Company, who have decided to concentrate at least part of their resources on the ZX market, had a range of books (nearly all of which were written in the UK) on display. Reston are also distributing a range of 16K ZX81 software written in Canada.

The bad news for Sinclair and Timex came from two areas. Texas unveiled their new computer, the TI 99/2. Selling for $99.00, this computer is a 4K black and whtie machine, housed in a solid and impressive-looking silver case with large discrete keys somewhat like those provided on calculators. Although it is only black and white (28 characters by 24 lines), it appears to outclass the ZX81 by miles. It is not particularly fast, but has a flicker-free display, which appears much faster than the ZX81 in Slow mode.

The other bad news comes from a new company, Venture Micro Inc. of Cupertino, California, who unveiled a very impressive $129.95 colour computer, the Humdinger. Smaller than a Spectrum, with the same sort of keys, the Humdinger has 13 (!) graphic modes with eight colours and a highest resolution of 256 x 192. This is the same as the Specturm, but in contrast to the Spectrum's 32 by 22 colour mask, the Humdinger has pixel-addressed colour. This, of course, consumes memory, but as the machine can be expanded to 64K, is not too much of a problem.

The basic Humdinger is supplied with 4K on board and the sound is superb. Four voices, with a range of five octaves, provide almost 'arcade quality' sound. The sound comes through the TV speaker.

Too Good To Be True

These specifications pale beside the fact that the Humdinger comes with interfaces for the following, all *built-in* on the standard machine: Centronics parallel printer port, serieal RS232 port with full handshake to connect to modem, terminal or network, joystick port and cartridge port for plug-in software. A standard disc operating system was said to be 'five weeks' away, along with a graphics tablet and an EPROM programmer.

Frankly, the Humdinger sounds too good to be true to me. But I bought one (not yet delivered, two weeks after the show) and so did Richard Turner, head of Artic Software. The company said they have no plans at present to launch the machine in the UK, but will be discussing doing so before the PCW show later in the year.

Presumably Richard Turner sees a potential software market for the machine. I'll report on the Humdinger, once I've put it through its paces, in a future issue of *ZX Computing*. Certainly it seems superb to me, an opinion shared by computer journalist Guy Kewney, who alerted me to the machine during a discussion in the Faire's press room.



The general feeling among computer dealers I spoke to in the US was that Timex have somehow missed the boat. Certainly, I saw only *one* TS 1000 on sale in the whole of San Francisco, and that was in a 'drug store', next to the watches and cheap transistor radios.

However, the ace up the Timex sleeve is the TS 1500, which is a ZX81, with 16K on board, furnished with a Spectrum-like keyboard. My spies indicate Timex will sell it for $99.00, exactly matched up against the TI 99/2. I don't know what this means for the TS 1000, which is supposed to have a list price of $99.00 (but which has been selling for as little as $57.50, according to some reports). I guess the TS 1000 will go on sale for $50.00, with the TS 1500 taking the higher level. The editor of one major US computer magazine told me that the difference between $70 and $130 (the current price of the Vic-20) was negligible to American consumers, so even at a low price, the TS 1000 may be battling uphill to get new buyers. We shall see!

Read All About It

A new American publication may be of interest to Timex/Sinclair 1000/ZX81 owners — priced at $5.00, it is called the Timex-Sinclair 1983 Directory.

In just over 90 double columned pages, the book covers such subjects as: where to find disc drives, RAM extensions, printers, modems, keyboards, interfaces, books, periodicals, programming aids, etc. Including photographs to illustrate the items, it describes special applications like control circuitry, enchanced graphics, voice generation, music synthesis, video inversion, light pens, joysticks, etc. There is also a guide to software available including everything from spreadsheets, word processors, data banks, engineering and design, to arcade and adventure games.

Those interested in this publication can find out more from the publishers, the E Arthur Brown Company, 1702 Oak Knoll Drive, Alexandria, MN 56308, USA.



Summer Holiday



Dolphin Activities Ltd was formed last year with the assistance of the Department of Industry's Education Unit and Information Technology Year '82 to operate Summer computer camps for young people. These proved so popular that Dolphin has developed a broader range of holidays and substantially increased capacity.

It is expected that 3,500 children will attend their summer camps this year, and to complement this, they are launching a unique family weekend at the centres where adults will be able to narrow the 'technology gap' and explore new interests with their children.

With enormous support from industry, Dolphin are able to provide over £100,000 of micros and interfacing equipment at each centre. The major computer support is from Sinclair Research Ltd who have provided 200 micro systems.

The courses are fun and informative and concentrate on three main areas: computing, robotics and psychobionics, while also aiming to teach beginners BASIC programming. Everyone has the opportunity to control and even build robots, and this is all combined with sporting and creative activities such as video film and cartoon making, windsurfing, go-carting and horse riding — all part of a well-planned week or weekend.

There is a wide variety of holiday options available and all are based in beautiful parts of the British countryside. For further details of these computing holidays, get in touch with Dolphin Activities Ltd, 68 Churchway, London NW1 1LT or telephone 01-387 5602. Have a nice Summer!

You may also like to note that as a special offer, should two children from one family wish to take a holiday at a Dolphin camp, or a child holidays for two weeks, Dolphin Activities will give them a free ZX81. Can't say fairer than that can you!

On The Run

Cambridge's second annual Festival Half-Marathon, sponsored by Sinclair Research for £5,000, is to be held on Sunday, 17 July. Starting at 10.00am from Parkers Piece, competitors, up to 2.000 atheletes, veterans and enthusiasts, will follow a two-lap course around the city by way of Fen Causeway, Grantchester and Trumpinton.

If you want to make a weekend of it, the Cambridge Festival itself begins on Saturday, 16 July. (If Dad doesn't want to drive you all the way there and back, you'd better play your trump card and tell him about the real ale tent!) You might even get a glimpse of Uncle Clive if, like last year, he enters the run himself.

And if you do manage to see him, you may see a wry smile on his face for if you had been reading some of the more up-market national press, you'll have noticed he was named The Guardian Young Businessman of the Year. The award, made this year for the 13th time (not that unlucky, we hope), celebrates a significant contribution to business at a personal, company and national level.



Clive running at last year's Half-Marathon.

Statistically Speaking

I thought you might be interested to hear about some new computing statistics compiled by the British Market Research Bureau based on their findings on a national survey of 2,000 households conducted in January and February of this year.

There are now nearly one million British households with their own computer, and this figure could double before the end of 1985. The leading makes, in terms of models owned by micro users, are the Sinclair ZX81 with 43% of the market and the ZX Spectrum with 14%. Over 40% of computers were bought as gifts and 57% of December 1982 sales were gift purchases (the ZX81 and ZX Spectrum accounted for a very large percentage of these).

The most frequently used software packages are the arcade games, although over 25% of households use them mainly for educational purposes.

An interesting statistic involved who actually uses the computers — mostly males, and in particular sons, are the users of the family home computer. The survey shows that 45% of users are aged 18 or under and only 12% are aged over 40. The survey also revealed that females have little interest in computers at all!

Citizen's Advice

Computers for Fun, Computers for Business (CFCB) have launched an enquiry bureau initially for Spectrum users. Among the services offered will be technical advice, programming advice and help with debugging your listings, and aid with actually listing your programs

The bureau charges £1.00 for each service required, and just 30p for the newsletter which will contain articles and letters of interest to all software and hardware users. Alternatively, a fee of £5.00 per annum entitles you to as many free enquiries as you like, as well as free copies of the newsletter. CFCB guarantee to return your money if they are unable to answer your query.

For further information or service contact CFCB, 610 Washwood Heath Road, Ward End, Birmingham B8 2HG. If you want to telephone, ask for Chris or Jean on 021-327 0984.

Software............... In Brief

● A new spreadsheet program called FlexiCalc is now available for the 48K Spectrum which will be of use in real planning situations. Priced at £9.95, FlexiCalc features control over the numbers of rows and columns of the spreadsheet, on-screen prompting, and full maths capability. For further details contact Saxon Computing, 3 St Catherines Drive, Leconfield, Beverley, Humberside or telephone 0401 50697.

● Flowchart Ltd have just released three new programs for the Sinclair Spectrum. IQ Test is a compelling way of testing your IQ, Joker is a fun program including a variety of jokes, puns, riddles and other graphical nonsense, and Home Budget is a package designed to enable a personal record to be created of estimated monthly expenses and income. For further details of these new programs get in touch with Flowchart Ltd, 62 High Street, Irthlingborough, Northants NN9 5TN or 'phone 0933 650073.

● The Invisible Man is an educational program released for use on the Spectrum which will help children, aged seven to 13, to learn more about co-ordinates and compass points. Priced at £5.95, there are three levels of difficulty to suit different age groups. For more information contact Chalksoft, Lowmoor Cottage, Tonedale, Wellington, Somerset TA21 OAL or telephone 082-347 7117.

● Another blockbuster from Melbourne House comes in the guise of Penetrator, in which your mission is to penetrate through four defence rings and blow up an illegal cache of neutron bombs — a virtually impossible feat! Enemy missiles and radar bases track your every movement as you bomb your way through the various caverns which you yourself can user-define. For further details contact Melbourne House Ltd, Glebe Cottage, Glebe House, Station Road, Cheddington, Leighton Buzzard, Bedfordshire LU7 7NA or 'phone 01-405 6347.

● Dietron is a new program for those of you wishing to establish your own personal nutritional diet. Based on the 48K Spectrum, this package lets you choose from 150 of the most popular kinds of food, showing the breakdown of calories, etc, helping you to lose weight. For more information contact Customdata, 20 Friars Quay, Colegate, Norwich NR3 1ES or telephone 0603 614812.

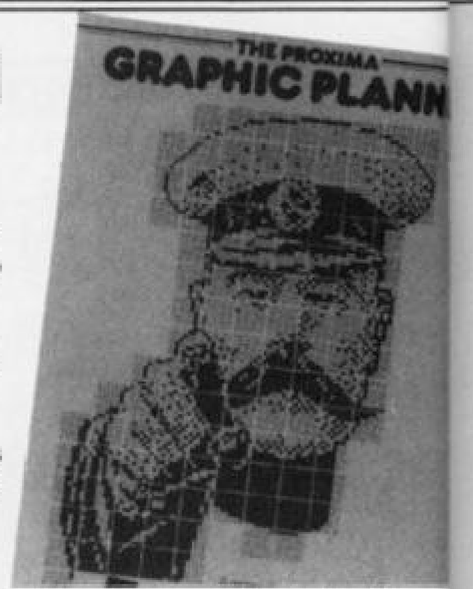
● Kuma Computers have released a new package called Map of the UK which is designed to run on the 48K Spectrum. Comprising a large scale map of all the United Kingdom plus the Republic of Eire, at any one time, a 75 by 60 mile area of the map can be viewed on-screen. For further information get in touch with Sassafras Software, 41 Skenfrith House, Commercial Way, London SE15 or 'phone 01-732 4777.

● Designed to operate on either the 16K or 48K computer comes Blind Alley, a simulation of a deadly duel, set deep in space. Priced at £4.95, you can find out more about the tape from Sunshine Books Ltd, Hobhouse Court, 19 Whitcomb Street, London WC2 7HF.

● Aimed at kindergarten children, this new package for the 16K Spectrum will help your child tell the time and distinguish between different coins and amounts of money. Available at £5.50, you can find out more about the program from Poppy Programs, Richmond House, Ingleton, Carnforth, Lancs LA6 3AN.

● A cassette for your 16K ZX81 has been released called High Resolution which contains three programs: Toolbox, Sine Wave and 3D Exponential Graphic Generator. Packaged by Computer Rentals Ltd, you get more details by writing to 140 Whitechapel Road, London E1 or 'phoning 01-247 9004.

Plan Ahead



New from Proxima Centauri Ltd comes the Proxima Graphic Planner, designed as a supplement and aid for all who are interested in defining their own computer graphics.

The Graphic Planner, priced at £3.50, is an A4, 80 page book which includes two pages of ideas and instructions for future use. This is followed by 12 pages containing over 300 pre-designed computer graphics which can be used immediately or used to spark your imagination. All the row values of the graphics have been calculated to save you time.

There are also two programs in BASIC which you can use to test your graphics, to calculate decimal values from binary, to POKE graphics to memory and to SAVE graphics to tape. Lastly, there are 64 pages, each printed with up to 48 eight by eight box arrays. These box arrays are arranged singly, in groups of two, four or 48, with column values pre-printed for your convenience. This latter facility is obviously useful if you are planning to include a number of user-definable graphics within one program and you wish to get some idea of what all the graphics will look like on one page.

So, if you want to free yourself from the keyboard and VDU, while still planning the graphics for your next epic program, why not get in touch with Proxima Centauri Ltd, 23 Denmark Street, London WC2H 8NA.

The Taxman Cometh

If you haven't got around to that most dreaded of all tasks, working out your tax for the year, here could be the answer for you.

Consisting of a series of programs on tape accompanied by a comprehensive manual, Microtax is claimed to be the first easy to use microcomputer system for completing tax returns. In addition to calculation of tax liabilities, Microtax also provides Spectrum users with all the details to be filled in on the 1983/4 tax return (which is the return of the income for the tax year 1982/3 and a claim for allowances for the tax year 1983/4). For those with their own printer, Microtax even provides a printout of all the relevant details so that they can be attached to the return.

Microtax was first developed by Tax and Financial Planning Ltd and took nine months of professional accountancy time and the equivalent of two person-years of program time to develop. The first Microtax package is an income tax system for 1982/3, but there will be a 1983/4, version of the program available in Autumn 1983. Also, in 1983, complimentary systems will be available dealing in more detail with the taxation of business and professional income and a third system to cover Capital Gains Tax.

Microtax is available for £24.94 (including postage and packing) from Microtax Ltd, Barratt House, 4th Floor, 7 Chertsey Road, Woking, Surrey GU21 5AB. Telephone enquiries can be made on 04862 20369.



Software Released

Quicksilva have been busy once again and released a new range of software for your 16K ZX81 and ZX Spectrum.

For the ZX81, there are five new packages: 3D Black Star, a fast 3D graphics game which places you deep in outer space; Ocean Trader, an adventure set in the 19th centruy; Damper and Glooper, a couple of fast machine code games; Cosmic Guerilla, a space game in which you are chased by a cluster of guerilla craft; and Pioneer Trail, a western adventure featuring 20 levels of play. 3D Black Star, Ocean Trader, Damper and Glooper, Cosmic Guerilla and Pioneer Trail are priced at £4.95, £3.95, £4.95, £3.95 and £3.95 repectively.

And if you think they sound exciting, there are three new packages for the ZX Spectrum. Two games for your 16K computer are Astroblaster, in which you are attacked by Cybirds, Meteors, plasma and goodness knows what else, and Frenzy, a game in which you stumble around a series of rooms taking special care not to step on the exploding pods or minelayers.



The last tape is called Word Processor and is designed to run on the 48K Spectrum. Featuring menu driven prompts, you can justify all your text both on-screen and on the printer. Astroblaster, Frenzy and Word Processor are priced at £4.95, £4.95 and £5.95 respectively.

I don't know whether you saw Nick Lambert of Quicksilva on the TV the other week, but Quicksilva are certainly proving the rags to riches tale. "We started off with a £200 bank overdraft and have never been in debt again. Next year, we predict a turnover of between £10 and £50 million" asserted the 31 year old ex-test engineer. Well, that certainly leaves a lot of room for error but remains an inspiration to software companies everywhere.

One Step Beyond

If you're tired of programming your computer with BASIC, you may like to take a chance with machine code. Machine code programming works a lot faster than BASIC and you'll save a lot of space in memory — all you have to do is learn the mysteries of machine code.

And before you give up, the City of London Polytechnic are offering an evening course in Z80 assembly language programming that is intended 'to strip the mystery from this fascinating subject and enable you to appreciate the innermost workings of your computer'.

At the end of the course, you should be able to write your own machine code programs — those that will stand on their own or perform utilities called from programs in BASIC.

The course takes place on 28th and the 30th of June, 1983, and is £15.00 per person. Application forms and further information are available from the Short Course Unit, City of London Polytechnic, 84 Moorgate, London EC2. Telephone applications should be made on 01-283 1030.

In Sickness . . .

A free-standing computer program which calculates Statutory Sick Pay (SSP) is now available from Hilderbay Ltd. use of the program requires no knowledge of computing or of the workings of SSP.

As you probably know if you are an employer or employee, Staturtory Sick Pay came into operation on 6 April, 1983. There is an employer's guide to SSP, but it's over 60 pages long! All employers must calculate and pay SSP to their sick employees, and then recover their payments by deducting them from their National Insurance Contributions — there is a fine of £200 plus £20 per day for failure to keep the required records.

The program, priced at £35 as part of an introductory offer, has been designed to run on the 48K Spectrum. The first time the user runs the program, it should help explain the necessary steps needed to be taken: a comprehensive list of exceptions are gone through; the program asks for details of the employee's pay and dates of sickness; and finally, the program computes and prints the SSP payable and other information required for future use.

For further information on this package, get in touch with Hilderbay Ltd, 8/10 Parkway, Regents Park, London NW1 7AA or telephone 01-485 1059.

On The Side

Microsphere is a new company set up to produce software, and three cassettes from their range are aimed at the ZX Spectrum.

The first, ZX-Sideprint, is a utility enabling printout to be produced sideways on a Sinclair Printer. As you can use print lines of any length, any columns of figures can be presented in a more readable 70 or 80 column format. The program is fully relocatable and includes a routine to actually do this job for you. The program is available for the ZX81 and ZX Spectrum and both are priced at £4.95 respectively.

The second program is aimed at the business market and is called Omnicalc. As its name suggests, it is a spreadsheet analysis program for the 48K Spectrum and is extremely useful for things mathematical. Written entirely in machine code, the program has been put together so that a first time user could easily pick it up; these are prompts for all inputs, verification of each character input, and a collection of meaningful error displays. Omnicalc is priced at £9.95.

The final cassette from the Microsphere stable is a games package for the 16K Spectrum. Priced at £4.95, the titles include at Crevasse and Hotfoot.

For further details of this software, contact Microsphere Computer Services Ltd, 72 Rosebery Road, London N10 2LA. Telephone enquiries can be made on 01-883 9411.

Hardware .... In Brief

• Cheetah Marketing Ltd have introduced two new RAM packs onto the market. Both models, professionally cased in a custom made unit, are designed to fit snugly into the back of the ZX81 ensuring that no program loss occurs through 'wobble'. The 16K and 64K versions are available priced at £19.95 and £44.75 respectively. Both packages are fully guaranteed for three months and comprehensive instructions are provided with each. For further information contact Cheetah Marketing Ltd, 359 The Strand, London WC2R OHS or 'phone 01-240 7939.

• A new 'quick reference' card for the ZX80 and ZX81 micros has been introduced by Elkan Electronics. The Nanos cards comprise a number of fold-up (accordian style) pages each with 10 panels of information. Written by Paul Nanos of Nanos Systems Corp., the cards are claimed to have all you need to program your ZX80/81. Priced at £3.50 each, you can find out more from Elkan Electronics, 11 Bury New Road, Prestwich, Manchester M25 6LZ or telephone 061-798 7613.

• DAZRAM (which stands for Database and ZX81 shared RAM) will connect a ZX81 and a Database together, greatly enhancing both products. It will enable the user to write machine code programs on the ZX81 and by a command from the keyboard, switch the program written to the Database to be played. By using the ZX81 and a new monitor program specially written for it, the DAZRAM will have more advanced editing facilities plus the ability to produce hard copy on the ZX Printer. Intended to help the newcomer to conquer machine code programming, you can find out more about DAZRAM from Voltmace Ltd, Park Drive, Baldock, Herts SG76EW or you could telephone 0462 894410.

• Fuller Micro Systems Ltd are boasting that they have sold more than 10,000 ZX81 keyboard and keyboard case conversions since exhibitiing the prototype way back in 1981. And now they have incorporated a sound expansion unit within the casings and are soon hoping to announce a fully re-designed keyboard for the Spectrum which will incorporate a space bar. ZX81 users can buy the conversion system for £29.95, the keyboard and case kit for £24.95 or the keyboard kit only for £14.95. The new re-designed keyboard with the space bar will cost £39.95. But that's not the end of the news from Fuller — they are also introducing the Fuller Box and the Fuller Orator. The Fuller Box offering amplification, joystick control and a sound synthesiser costs £29.95, and the Fuller Orator, which translates typed-in words from the keyboard into spoken words and sentences is priced at £39.95. For further details of what Fuller have in store for you in 1983, why not contact Fuller Micro Systems, Sweeting Street, Liverpool.

• The ROM-81 is a memory expansion unit for the ZX81 enabling the user to read useful routines and commonly used information which is stored in UV erasable, programmable Read Only Memory. As the most popular EPROMs have a maximum access time of around 450 nanoseconds, there is a special 'Wait State' circuit in the ROM-81 which automatically requests that the CPU in the ZX81 waits until the data has been read. Available in a black ABS case, the ROM-81 device is designed to plug onto the back of the ZX81 allowing further expansions to be made. It is supplied with comprehensive user notes which provide programs for data retrieval. For more information on the ROM-81, get in touch with Cambridge Microelectronics Ltd, 1 Milton Road, Cambridge CB4 1UY

Window Shopping?

Do you need any help with PLOT, DRAW and CIRCLE on your Spectrum? If you do, Victa Ceramics may just have come out with the perfect idea for you.

Comprising a white plastic base sheet, a clear plastic offset co-ordinate window, a clear plastic calibrated window and a clear plastic window mask with a plot sight in each corner, the Victagraph also includes a test design, comprehensive instructions and a number of sticky patches. Using the various window masks over a test design, the mask will hide all the unwanted co-ordinates leaving you with the required figures — making a difficult task a lot easier!

For more information on the Victagraph, get in touch with Victa Ceramics, 6a Bow Street, Rugeley, Staffordshire or telephone 08894 2426. The Victagraph package is priced at £7.50

Finger Fever

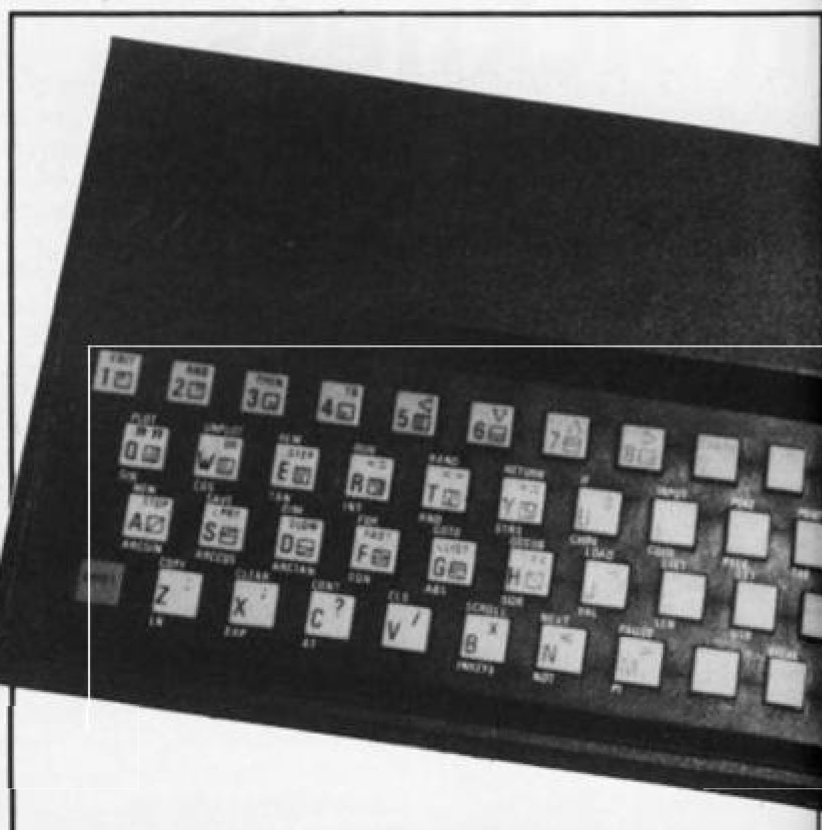
Simple ideas are always the best ideas, and this product is no exception.

Consisting of a black ABS resin case, this push button keyboard for the ZX81 is fixed to the computer via a number of adhesive pads. Once this is done, you have a much improved keyboard for your computer.

There is little problem locating the keyboard in the correct place over the keyboard, you just have to make sure that the whole unit is central. Once attached, you have a ZX81 with a keyboard made up of raised keys, each key having a travel of about three to four millimetres — which is enough to make sure you have a positive feel.

The spring in the keys is provided by the base plate, each key pressing down onto a triangular tab which is part of the moulding of the base. This in turn causes the raised peg on the other side of the tab to make contact with the surface of the ZX81's keyboard.

The Push Button keyboard is priced at £9.95 and you can find out more about this product from File 60, FREEPOST, London W9 2BR.



Would you Adam and Eve it?

Stephen Adams has come up with a number of devices for the Sinclair range of computers. First up is the Straight adaptor, which converts the ZX Spectrum to the same expansion port as the ZX81, but does not do any address conversion. Thus, you can obtain a full 64K of addresses when the printer is in use. It may also be used to isolate some of the dangerous voltages involved when using I/O ports. The device, like the rest of the range of adaptors, plugs into the expansion port at the back of the Spectrum and does not require any special instructions or machine code to use it.

The Eve adaptor is available for those people who own the 48K Spectrum, allowing them to access the wide range of ZX81 add-on devices now on the market. However, the Eve adaptor will only work for devices which work in the 0-16K section of the ZX81's memory map and it will not allow you to add more than 48K RAM. The Adam adaptor, on the other hand, has been developed for the 16K Spectrum and simply allows users to add on the Sinclair (or compatible) 16K RAM Pack doubling your memory at a stroke.

An update of the Adam adaptor is the aptly-named Adam II adaptor. This device allows the use of two sets of peripherals to be added at the same time on the 16K Spectrum. Thus, you could have a 16K RAM pack and any device which was contained in the 0-16K area of the ZX81's memory map such as ports and EPROM programmers (but not ZX81 EPROM programmers), or a battery backed RAM.

All the adaptors are available from Stephen Adams for £9 including VAT, postage and packing.

But before you rush off, Stephen has also developed what he claims to be the only programmable tape controller available for the ZX81 and Spectrum. The device can be used to select which tape recorder lead is required for SAVEing or LOADing; thus, you don't have to pull out the cassette leads causing wear on the sockets of the computer. The unit also turns on and off the cassette motor under program control and provides a minimum of five latched data outputs for controlling external devices. The outputs can also be used for providing up to 32 different devices at one memory or I/O address.

Designated the RZ1 tape controller, the device has the ability to work on any machine as it uses no I/O address or RAM memory location. The price of the RZ1 is £20 including VAT, postage and packing.

If you want more information on any of Stephen's hardware add-ons, you can write to him at 1 Leswin Road, London N16 7NL or 'phone him on 01-254 1869.

Getting Into Print

If you've felt the frustrations of utilising the ZX Printer or would like to orient the Spectrum more towards the business side, you may like to know of the Kempston Centronics interface which can be used to link your computer up to Centronics type printers.

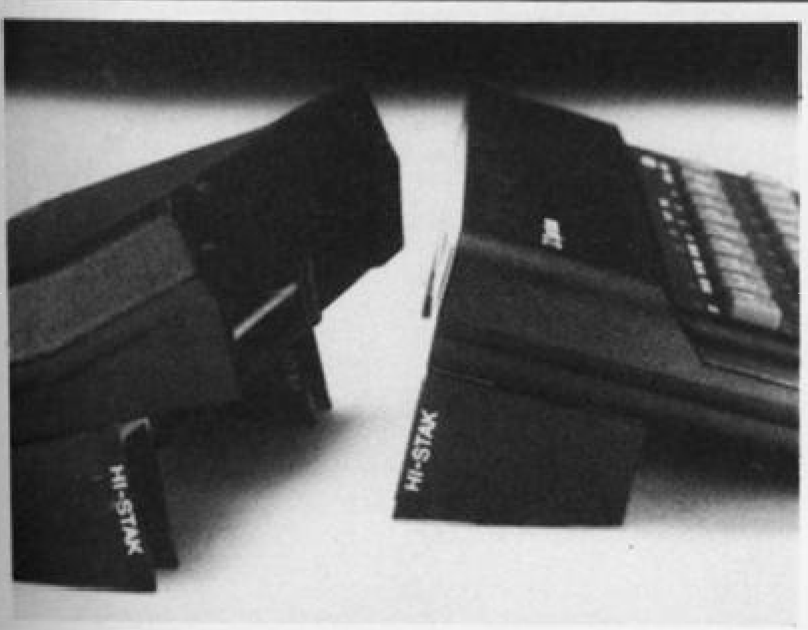
A major feature of this package is its recognition of LLIST and LPRINT, which means that you can list directly from the Spectrum and also get printout direct from listings (BASIC only) without the need for special user calls. It is also possible to send out control codes to the printer giving the facility of different characters, for example, condensed, expanded, etc.

The interface is supplied cased and ready to use by simply plugging it into any Centronics type printer, such as the Seikosha 100 A, the Oki Microline 80, and all Epsons including the MX-80 F/T111, etc.

Complete with connecting lead, the Centronics interface is priced at £45.00 plus £1.00 postage and packing, which includes a 12 month guarantee. For further information get in touch with Kempston (Micro) Electronics, 180a Bedford Road, Kempston, Bedford MK42 8BL.



On The Level?



Hi-Stak is a new add-on which has been specifically designed to improve the usability of a wide range of personal computers, including the ZX81 and ZX Spectrum, by raising the back to a calculated level. Fundamental ergonomics reveal that this makes the keys more comfortable to operate, and their printed surfaces that bit easier to read.

A simple idea, the Hi-Stak can be instantly applied, and comprises two precision injection moulded ABS ramps with built-in rubber feet, self adhesive tops and easy to follow instructions.

So, if you want to add that little professional touch to your system, have a look at the Hi-Stak available for £3.95. For more information contact Warp Factor Eight, 6 Pelham Road, Braughing, Ware, Herts SG11 2QU. Telephone enquiries can be made on 0920 821841.

Monster pit



Dare you delve into the dark recesses of the Monster Pit? A great adventure for you and your ZX81 from Jim Enness.

Whilst searching a ruined tower you are attacked by a Screech Bat which knocks you into a pit containing caves, passages and stairs, etc. Many of the caverns contain monsters; in fact, they are all over the place and roam the passages as well! Some caves are empty while others contain treasures in the form of gold, spells and hit points.

The object is, of course, to make your way out, killing any monsters you meet and carrying all the gold you can. You have fallen twenty levels down and to get out you must find stairs up and then the exit.

If you are faced with a monster, you may 'Combat' it using your initial 100 hit points; you may 'Cast a spell' on it which throws a random number 0 to 100 hit points at the monster; or you may 'Retreat' which takes you down to the level below – not the way you want to go, but it is one way of avoiding a nasty incident. If a monster has more hit points than you and should attack then you lose the game.

Treasure hunt

Various treasures in the caves boost your hit points and spells as you find them. Gold has a considerable influence on your score at the end of the game – if you make it out alive that is! Screech Bats reside all over the roofs but generally leave you alone. They are, however, attracted by the movement of large amounts of gold and sometimes attack if you collect a small fortune of 150 pieces of gold or more. You can't kill the Screech Bats, they simply knock you down a floor, take your gold, half your hit points and break half your spells – not very nice.

If you press the key 'S', when asked 'What Direction?' a status chart tells you your strength and position; Newline clears the chart leaving the 'What Direction?' question to be answered. If you type in the letter 'A', the program will SAVE and then continue where it left off. The game can, therefore, be continued at another time.

All key entries are single letter ones, eg 'L' for left, or 'F' for forward, etc, using the INKEY$ mode. 'RUN' starts the game or a new game.

There are many hazards I have not mentioned, but you will no doubt find them if you think you can get out alive!!! By the way, the Screech Bats also reside in the tower above the pit and automatically attack anyone attempting to go above the ground level.

Programming notes

The program of this game is fairly self-explanatory with the aid of the REM statements. The main program at line 400 branches to the various subroutines to select the random possibilities of the next move and the display.

The main input routine at line 430 converts the value of your input to the relevant number which is then used to check the contents of the dimensioned B variable. (B is dimensioned into 18 and holds six pieces of information on each of the three directions updated by the main program.)

The variables used in the program are as follows:

P – The highest points scored so far.
G – The amount of gold.
S – The number of spells.
H – The number of hit points.
L – The level of the pit.
K – The number of monsters killed.
E – The end level (allows exit to be used on level 1 or level 0).

F is used to indicated whether to set up a new field at the beginning of the main program. C,R,W,X,Z and Z$ are used as temporary variables for immediate data.

Once the program has been RUN, it is a good idea to use the SAVE routine at the beginning of a new game to tape out. RUNning the program again will lose the 'Highest score so far' variable, and it is always an enticement to see if you can improve on the last effort.

```
  6 LET P=0
  8 GOSUB 1750
  9 REM SET WORDS
 10 DIM A$(135,10)
 12 LET A$(1)="CAVE ENTR."
 14 LET A$(2)="   DOOR   "
 16 LET A$(3)="BLANK WALL"
 18 LET A$(4)="PASSAGEWAY"
 20 LET A$(5)="STAIRS UP."
 22 LET A$(6)="STEPS DOWN"
 24 LET A$(7)="   EXIT   "
 26 LET A$(8)="          "
 28 LET A$(9)="   LEFT   "
 30 LET A$(10)=" FORWARD. "
 32 LET A$(11)="  RIGHT.  "
 34 LET A$(12)="SALAMANDER"
 36 LET A$(13)="RED ZOMBIE"
 38 LET A$(14)="BEHOLDER I"
 40 LET A$(15)="BEHOLDER V"
 42 LET A$(16)="BEHOLDER X"
 44 LET A$(17)="HELL HOUND"
 46 LET A$(18)="BLUE ETTIN"
 48 LET A$(19)="GREY ETTIN"
 50 LET A$(20)="PIT WRAITH"
 52 LET A$(21)="GARGOYLE I"
 54 LET A$(22)="GARGOYLE V"
 56 LET A$(23)="GARGOYLE X"
 58 LET A$(24)="FIRE ETTIN"
 60 LET A$(25)="CLAY ETTIN"
 62 LET A$(26)="IRON ETTIN"
 64 LET A$(27)="HILL ETTIN"
 66 LET A$(28)="BASILISK I"
 68 LET A$(29)="BASILISK V"
 70 LET A$(30)="BASILISK X"
 72 LET A$(31)="ZOMBIE RAT"
 74 LET A$(32)="WRAITH ORC"
 76 LET A$(33)="PINK GHOST"
 78 LET A$(34)="ICE WRAITH"
 80 LET A$(35)="MIND GHOST"
 82 LET A$(36)="FIRE GHOST"
 84 LET A$(37)="CLAY GHOST"
 86 LET A$(38)="IRON GHOST"
 88 LET A$(39)="MIND HYDRA"
 90 LET A$(40)="PIT HYDRA"
 92 LET A$(41)="CLAY HYDRA"
 94 LET A$(42)="IRON HYDRA"
 96 LET A$(43)="RED MEDUSA"
 98 LET A$(44)="MEDUSA ORC"
100 LET A$(45)="ICE MEDUSA"
102 LET A$(46)="PIT MEDUSA"
104 LET A$(47)="RED FLAYER"
106 LET A$(48)="BLUE MUMMY"
108 LET A$(49)="CLAY MUMMY"
110 LET A$(50)="IRON MUMMY"
112 LET A$(51)="MINOTAURUS"
114 LET A$(52)="MINOTAURAK"
116 LET A$(53)="MINOTAURIC"
118 LET A$(54)="PIT FLAYER"
120 LET A$(55)="ICE FLAYER"
122 LET A$(56)="WERE-WOLF"
124 LET A$(57)="WERE-BOAR"
126 LET A$(58)="WERE-BEAR"
128 LET A$(59)="WERE-RAT"
130 LET A$(60)="WERE-WORM"
132 LET A$(61)="BLACK-WORM"
134 LET A$(62)="WHITE WORM"
136 LET A$(63)="STONE WORM"
138 LET A$(64)="BONE-WORM"
140 LET A$(65)="UMBER HULK"
142 LET A$(66)="STONE HULK"
144 LET A$(67)="GREEN HULK"
146 LET A$(68)="CLOUD HULK"
148 LET A$(69)="STORM HULK"
150 LET A$(70)="HOBGOBLIN"
152 LET A$(71)="HOBGOBELOW"
154 LET A$(72)="MANTACORE"
```





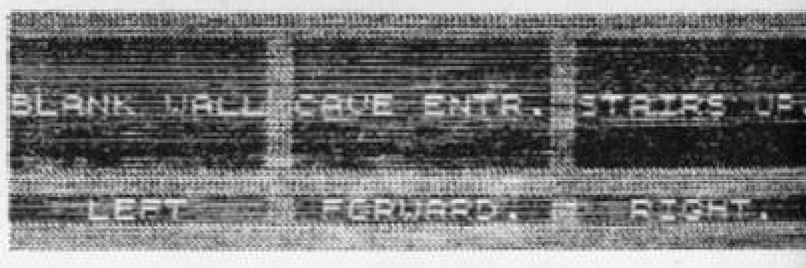
```
156 LET A$(73)="BAALZEBUL"
158 LET A$(74)="TELEMENTAL"
160 LET A$(75)="PIT FIEND"
162 LET A$(76)="HORNED ORC"
164 LET A$(77)="BARBED ORC"
166 LET A$(78)="RED GERYON"
168 LET A$(79)="JUBILEX I"
170 LET A$(80)="JUBILEX II"
172 LET A$(81)="DEMOGORGON"
174 LET A$(82)="RED ORCUS"
176 LET A$(83)="BLUE ORCUS"
178 LET A$(84)="GREY ORCUS"
180 LET A$(85)="FIRE ORCUS"
182 LET A$(86)="WILD TROLL"
184 LET A$(87)="HELL TROLL"
186 LET A$(88)="FIRE TROLL"
188 LET A$(89)="BOAR TROLL"
200 LET A$(90)="BLUE TROLL"
202 LET A$(91)="WILD DEMON"
204 LET A$(92)="BLUE DEMON"
206 LET A$(93)="RED DEMON"
208 LET A$(94)="PIT DEMON"
210 LET A$(95)="FIRE DEMON"
212 LET A$(96)="ICE DEMON"
214 LET A$(97)="BEAR DEMON"
216 LET A$(98)="CLAY DEMON"
218 LET A$(99)="GREY DEMON"
220 LET A$(100)="IRON DEMON"
222 LET A$(101)="WILD ETTIN"
224 LET A$(102)="PIT ETTIN"
226 LET A$(103)="ICE ETTIN"
228 LET A$(104)="MIND ETTIN"
230 LET A$(105)="HELL ETTIN"
232 LET A$(106)="WILD GHOST"
234 LET A$(107)="RED GHOST"
236 LET A$(108)="PIT GOLEM"
238 LET A$(109)="ICE GOLEM"
240 LET A$(110)="FLESH RAT"
242 LET A$(111)="MIND DEVIL"
244 LET A$(112)="FLESH HULK"
246 LET A$(113)="GOLD COINS"
248 LET A$(114)="+ A SILVER"
250 LET A$(115)=" SWORD"
252 LET A$(116)="+ A MAGIC"
254 LET A$(117)="LANCE"
256 LET A$(118)="+ A SUIT"
258 LET A$(119)="OF ARMOUR"
260 LET A$(120)="+ A SPECIA"
262 LET A$(121)="L ARTEFACT"
264 LET A$(122)="+ AN ANCIE"
266 LET A$(123)="NT SCROLL"
268 LET A$(124)="+ A SECRET"
270 LET A$(125)=" POTION"
272 LET A$(126)="+ SOME MOR"
274 LET A$(127)="E SPELLS"
276 LET A$(128)="+ A WISDOM"
278 LET A$(129)=" BOOK"
280 LET A$(130)="+ A GOLDEN"
282 LET A$(131)=" RING"
284 LET A$(132)="+ A SPELL"
286 LET A$(133)="LUCKSTONE"
299 REM SET UP
300 LET C=0
302 LET E=6
304 LET L=20
306 LET K=0
308 LET H=100
310 LET F=0
312 LET S=10
314 LET G=0
316 GOTO 403
400 REM MAIN PROGRAM
401 IF F=1 THEN GOTO 426
402 IF INT (RND*200)<5 THEN GOSUB 1000
403 DIM B(18)
404 IF L=1 THEN LET E=7
405 IF L<0 OR G>150 AND S>2 AND INT (RND*100)<8 THEN GOSUB 750
406 FOR Z=1 TO 3
408 LET B(Z)=INT (RND*E)+1
410 IF B(Z)<3 THEN GOSUB 1100
412 NEXT Z
414 IF B(1)=3 AND B(2)=3 AND B(3)=3 THEN GOTO 406
416 CLS
417 GOSUB 506
418 GOSUB 500
420 PRINT A$(B(1));"▒";A$(B(2));"▒";A$(B(3))
422 GOSUB 500
423 GOSUB 506
424 PRINT A$(9);"▒";A$(10);"▒";A$(11)
425 GOSUB 506
426 PRINT
427 LET F=0
428 PRINT "WHAT DIRECTION? ";
430 LET Z=0
432 IF INKEY$="" THEN GOTO 432
434 LET Z$=INKEY$
435 IF Z$="A" THEN GOTO 2000
436 IF Z$="L" THEN LET Z=1
438 IF Z$="F" THEN LET Z=2
440 IF Z$="R" THEN LET Z=3
441 IF Z$="S" THEN GOSUB 550
442 IF Z<1 THEN GOTO 432
444 PRINT Z$
446 IF B(Z)=6 THEN LET L=L+1
448 IF B(Z)=5 THEN LET L=L-1
450 IF B(Z)=3 THEN GOSUB 600
452 IF B(Z)=7 THEN GOSUB 700
454 IF B(Z)=2 THEN GOSUB 800
456 IF B(Z)=1 THEN GOSUB 900
457 IF INT (RND*100)>90 THEN GOSUB 1300
458 GOTO 400
499 REM SET DISPLAY AND NO.S
500 PRINT A$(8);"▒";A$(8);"▒";A$(8)
502 PRINT A$(8);"▒";A$(8);"▒";A$(8)
504 RETURN
506 PRINT "▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
508 RETURN
510 LET R=INT (RND*101)+12
512 RETURN
514 PRINT
516 PRINT "THE CAVE CONTAINS ";
518 PRINT "A";
520 IF A$(R,1)="I" THEN PRINT "N";
522 PRINT " ";A$(R);
524 RETURN
526 FOR X=11 TO 21
528 PRINT AT X,0;"                                "
530 NEXT X
532 PRINT AT 11,0;" "
534 RETURN
536 LET W=INT (RND*101)+12
538 RETURN
540 LET W=(INT (RND*16)+1)*2
542 RETURN
549 REM STATUS
550 PRINT "STATUS",
552 GOSUB 526
554 PRINT "YOU HAVE KILLED ";K;" MONSTERS"
556 PRINT
558 PRINT "YOU ARE ON LEVEL ";L,,,
560 PRINT "YOU HAVE ";H;" HIT P
```

```
OINTS",,,
 562 PRINT "YOU HAVE ";S;" SPELL
S",,,
 564 PRINT "AND YOU HAVE ";G;" G
OLD PIECES ";
 556 IF INKEY$="" THEN GOTO 566
 570 GOSUB 526
 571 PRINT AT 10,16;"          ";AT
 10,16;
 572 RETURN
 599 REM WALL
 600 GOSUB 526
 602 PRINT "NO-ENTRY....WALL UNS
TABLE",,,
 604 LET F=1
 606 GOSUB 536
 608 IF W>50 THEN RETURN
 610 PRINT "IT SHAKES..ROOF CRUM
BLES..ROCKS",,,"FALL..KNOCK YOU
TO THE GROUND",,,"TAKE ";W;" HIT
 POINTS DAMAGE."
 612 LET H=H-W
 614 RETURN
 699 REM END SCORE
 700 CLS
 702 PRINT "WELL DONE,YOU GOT OU
T ALIVE."
 704 PRINT
 706 PRINT "YOU KILLED ";K;" MON
STERS.",,,
 708 PRINT "YOU HAVE ";G;" GOLD
COINS",,,
 710 PRINT "YOU SCORED ";G*K+S+H
;" POINTS.",,,
 712 IF P<G*K+S+H THEN LET P=G*K
+S+H
 714 PRINT "BEST SCORE SO FAR ";
P,,,,
 716 PRINT "ANOTHER GAME?"
 718 IF INKEY$="" THEN GOTO 714
 720 GOTO 8
```



```
 749 REM SCREECH BAT
 750 CLS
 752 PRINT
 754 PRINT "  A SCREECH BAT DESC
ENDS ON YOU ",,"KNOCKS YOU TO TH
E FLOOR AND",,,"TAKES ALL YOUR G
OLD,GIVING YOU",,,INT (H/2);" HI
T POINTS DAMAGE:BREAKING"
 756 PRINT
 758 PRINT INT (S/2);" OF YOUR S
PELLS:LEAVING YOU",,,"ON LEVEL "
;L+1;"."
 759 LET G=0
 760 LET H=H-INT (H/2)
 762 LET S=S-INT (S/2)
 768 LET L=L+1
 770 PRINT
 772 PRINT "    (HIT NEW-LINE TO
CONTINUE)"
 774 IF INKEY$="" THEN GOTO 774
 776 RETURN
 799 REM DOOR ENTRY
 800 GOSUB 526
 801 PRINT "DOOR...O-OPEN,L-LIST
EN ";
 802 IF INKEY$="" THEN GOTO 802
 804 LET Z$=INKEY$
 806 IF Z$="L" THEN GOTO 811
 808 IF Z$="O" THEN GOTO 898
 810 GOTO 802
 811 PRINT Z$,,
 812 PRINT "YOU HEAR ";
 814 IF B(Z+3)>0 THEN PRINT A$(B
(Z+3));"S MAKING",,,"NOISES";
 816 IF B(Z+3)=0 THEN PRINT "NOT
HING.";
 818 PRINT "...NOW WHAT?",,,"G-G
O IN,H-HELP,L-LEAVE ";
 820 IF INKEY$="" THEN GOTO 820
 822 LET Z$=INKEY$
 824 IF Z$="G" THEN GOTO 850
 826 IF Z$="H" THEN GOTO 950
 828 IF Z$="L" THEN GOTO 832
 830 GOTO 820
 832 PRINT Z$;"-CHICKEN",,
 838 LET F=1
 840 IF INT (RND*100)<10 THEN GO
SUB 1000
 842 RETURN
 849 REM ENTRY CONTENTS
 850 GOSUB 526
 852 PRINT AT 10,0;
 854 IF B(Z+3)>0 THEN LET R=B(Z+
3)
 856 IF B,(Z+3)>0 THEN GOSUB 516
 857 IF B(Z+3)>0 THEN PRINT
 858 IF B(Z+3)=0 THEN PRINT "A M
ONSTER-LESS CAVEN"
 860 PRINT
 862 IF B(Z+6)>0 THEN PRINT "WIT
H ";B(Z+6);" ";A$(113),,,
 864 IF B(Z+9)>0 THEN PRINT A$(B
(Z+9)+100);A$(B(Z+9)+101),,,
 866 IF B(Z+12)>0 THEN PRINT A$(
B(Z+12)+100);A$(B(Z+12)+101),,,
 868 IF B(Z+15)>0 THEN PRINT A$(
B(Z+15)+100);A$(B(Z+15)+101),,,
 870 IF B(Z+3)>0 THEN GOSUB 1314
 871 IF B(Z+3)>0 AND C=1700 THEN
 GOTO 838
 872 LET G=G+B(Z+6)
 874 FOR X=9 TO 15 STEP 3
 876 IF B(Z+X)>0 AND B(Z+X)<25 T
HEN LET H=H+30
 878 IF B(Z+X)>23 THEN LET S=S+1
 880 NEXT X
 882 RETURN
 897 REM CAVE CONTENTS
 898 PRINT Z$
 900 GOSUB 526
 902 IF B(Z+3)>0 THEN LET R=B:Z+
3)
 904 IF B(Z+3)>0 THEN GOSUB 516
 905 LET W=0
 906 FOR X=6 TO 15 STEP 3
 908 IF B(Z+3)>0 AND B(Z+X)>0 TH
EN LET W=2
 910 IF B(Z+3)=0 AND B(Z+X)>0 TH
EN LET W=1
 912 NEXT X
 914 IF W=2 THEN PRINT ,,,"AND T
REASURE";
 916 IF W=1 THEN PRINT "THE CAVE
 CONTAINS TREASURE",,,,
 918 IF B(Z+3)=0 AND W=0 THEN PR
INT "THE CAVE IS EMPTY";
 919 IF B(Z+3)>0 AND W=0 THEN PR
INT ,,
 920 GOTO 818
 949 REM HELP
 950 PRINT Z$
 951 GOSUB 526
 952 IF B(Z+3)>0 THEN PRINT "THE
 ";A$(B(Z+3));" HAS ";B(Z+3)-12;
" HIT POINTS"
 954 IF B(Z+3)=0 THEN PRINT "NO
MONSTERS?"
 956 PRINT
 957 PRINT ,
 958 GOTO 818
 999 REM FLOOR CAVE-IN
1000 GOSUB 526
```

```
1001 LET F=0
1002 GOSUB 536
1004 PRINT "THE FLOOR CAVES IN A
ND YOU FALL",,,"DOWN A ";INT (W/
2);" FOOT PIT TO LEVEL",,,
1006 LET L=L+INT (W/12)
1008 PRINT """";L;"""" ";
1010 GOSUB 536
1012 IF W>95 THEN PRINT " AND IN
TO A POOL OF ACID",,,"TAKE ";H;"
 HIT POINTS DAMAGE.",,,
1014 IF W>95 THEN LET H=0
1016 IF W<40 THEN PRINT " AND ON
TO SOME SPIKES",,,"TAKE ";INT (H
/2);" HIT POINTS DAMAGE",,,
1018 IF W<40 THEN LET H=H-INT (H
/2)
1020 RETURN
1099 REM SET ROOM CONTENTS
1100 FOR X=3 TO 6 STEP 3
1102 GOSUB 536
1104 IF W>70 THEN LET B(Z+X)=W
1106 GOSUB 536
1108 IF B(Z+X)>1 THEN LET B(Z+X)
=W
1109 NEXT X
1110 FOR X=9 TO 15 STEP 3
1112 GOSUB 536
1113 IF W>75 THEN LET B(Z+X)=W
1114 GOSUB 540
1116 IF B(Z+X)>0 AND W>12 THEN L
ET B(Z+X)=W
1117 IF B(Z+X)>0 AND W<13 THEN L
ET B(Z+X)=0
1118 NEXT X
1120 RETURN
1299 REM MONSTER APPEARS
1300 GOSUB 526
1302 GOSUB 510
1304 GOSUB 518
1306 PRINT " ARRIVES"
1308 PRINT
1310 PRINT "WHAT NOW?"
1312 PRINT
1314 PRINT "C-COMBAT S-SPELL CAS
T R-RETREAT"
1316 LET C=0
1318 IF INKEY$="" THEN GOTO 1318
1320 LET Z$=INKEY$
1322 IF Z$="C" THEN LET C=1500
1324 IF Z$="S" THEN LET C=1600
1326 IF Z$="R" THEN LET C=1700
1328 IF C<1 THEN GOTO 1318
1329 PRINT Z$
1330 GOSUB C
1332 IF C<1 THEN GOTO 1316
1334 RETURN
1499 REM BATTLE RESULTS
1500 GOSUB 526
1502 PRINT "YOU HAD A TERRIFIC B
ATTLE WITH"
1504 PRINT
1506 PRINT "THE ";A$(R);
1508 IF H<(R-12) THEN PRINT " BU
T HE KILLED YOU",,"AND TOOK ALL
YOUR TREASURE."
1510 IF H<(R-12) THEN GOTO 711
1512 PRINT " AND YOU KILLED",,,"
HIM."
1514 LET K=K+1
1516 LET H=H+12-R
1518 RETURN
1520 PRINT "ER..SORRY..YOUR OUT
OF SPELLS."
1522 PRINT
1524 PRINT "NOW WHAT?"
1526 LET C=0
1528 RETURN
1599 REM SPELL RESULTS
1600 GOSUB 526
1602 GOSUB 536
1604 IF S<1 THEN GOTO 1520
1606 LET S=S-1
1608 IF W>R THEN PRINT "THE NOW
SPELL BLASTED",,,A$(R);" LIES DE
AD ON THE",,,"FLOOR."
1610 IF W>R THEN LET K=K+1
1612 IF W>R THEN RETURN
1614 PRINT "THE ";A$(R);" IS ANG
RY HE",,,"ADVANCES...";
1616 IF W<27 THEN PRINT ".HE ATT
ACKS.........."
1618 IF W<27 THEN GOTO 1502
1620 PRINT "WHAT NOW?"
1622 LET C=0
1624 RETURN
1700 GOSUB 536
1701 GOSUB 526
1702 IF W<40 THEN PRINT "OH..BAD
 LUCK HE ATTACKS",,,
1704 IF W<40 THEN GOTO 1502
1706 PRINT "..CHICKEN",,,
1708 LET F=0
1709 LET L=L+1
1710 RETURN
1749 REM INTRO
1750 CLS
1751 PRINT "         PIT WITH MONS
TERS",,,,,,,,,"BY JIM ENNESS"
1752 IF INKEY$="" THEN GOTO 1752
1753 CLS
1754 PRINT "YOU FALL DOWN A 150
FOOT PIT",,,"THAT CONTAINS TREAS
URE AND MANY",,,"HAZARDS. THE OB
JECT OF THE GAME",,,
1755 PRINT "IS TO MAKE YOUR WAY
UP THE STEPS",,"THROUGH THE MAZE
 OF CAVES TO THE",,"EXIT. PRESS
KEY ""S"" WHEN ASKED",,,
1756 PRINT "WHAT DIRECTION? WILL
 GIVE YOUR",,,"STATUS-THEN ANY K
EY TO CONTINUE.",,
1758 PRINT "IN COMBAT YOU USE HI
T POINTS TO",,,"KILL MONSTERS-SP
ELL CASTING IS",,,"OFTEN SUCCESS
FUL(NOT ALWAYS)."
1760 IF INKEY$="" THEN GOTO 1760
1761 CLS
1762 PRINT "OH..BY THE WAY, BEST
 OF LUCK....",,"........YOU""LL
 NEED IT."
1764 RETURN
2000 SAVE "PIT WITH MONSTERS"
2002 GOTO 400
5000 SAVE "PU[illegible]"
5002 RUN
```



WHAT DIRECTION? STATUS

YOU HAVE KILLED 5 MONSTERS

YOU ARE ON LEVEL 21

YOU HAVE 194 HIT POINTS

YOU HAVE 0 SPELLS

AND YOU HAVE 375 GOLD PIECES

A TYPICAL DISPLAY SHOWN WHEN

"S" IS PRESSED WHEN ASKED A

DIRECTION.

A sample screen illustration from the program, Monster pit, showing the player's status.

Adding interest to your programs — part two

Tim Hartnell continues in his quest to illustrate just how a simple listing can be transformed into a program you can be proud of.

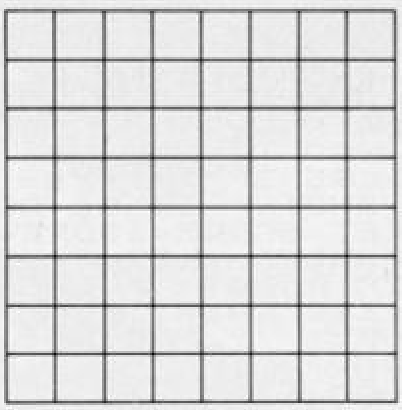
First off, I hope you had a lot of fun with the three listings, Programs 1, 2 and 3, that I introduced you to last issue. We now take off the kid gloves and take a PEEK at Program 4. As you can see from a quick glance, the listing looks considerably different from the listings we've examined up to now.

At the very least, you must admit that line 40 now looks extremely strange. This line is where a user-defined graphic (which, believe it or not, *does* look like a duck when the program is RUNning) takes the place of the randomly-chosen letters. No matter how hard you look, you will not find anything that looks like that little duck on the keyboard. So, where has it come from?

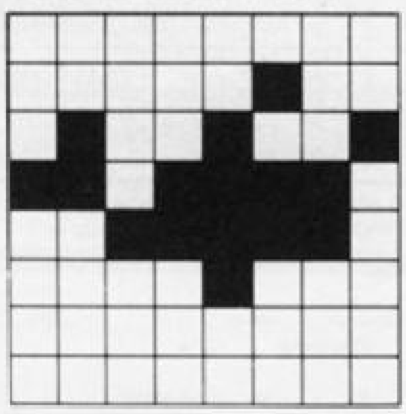
Duck defining!

The duck has been 'user-defined'. User-definable graphics are one of the really advanced features of the Spectrum, and although certain other computers (which shall remain nameless) have similar features, few (if any) are as simple to use as the one on your Spectrum.

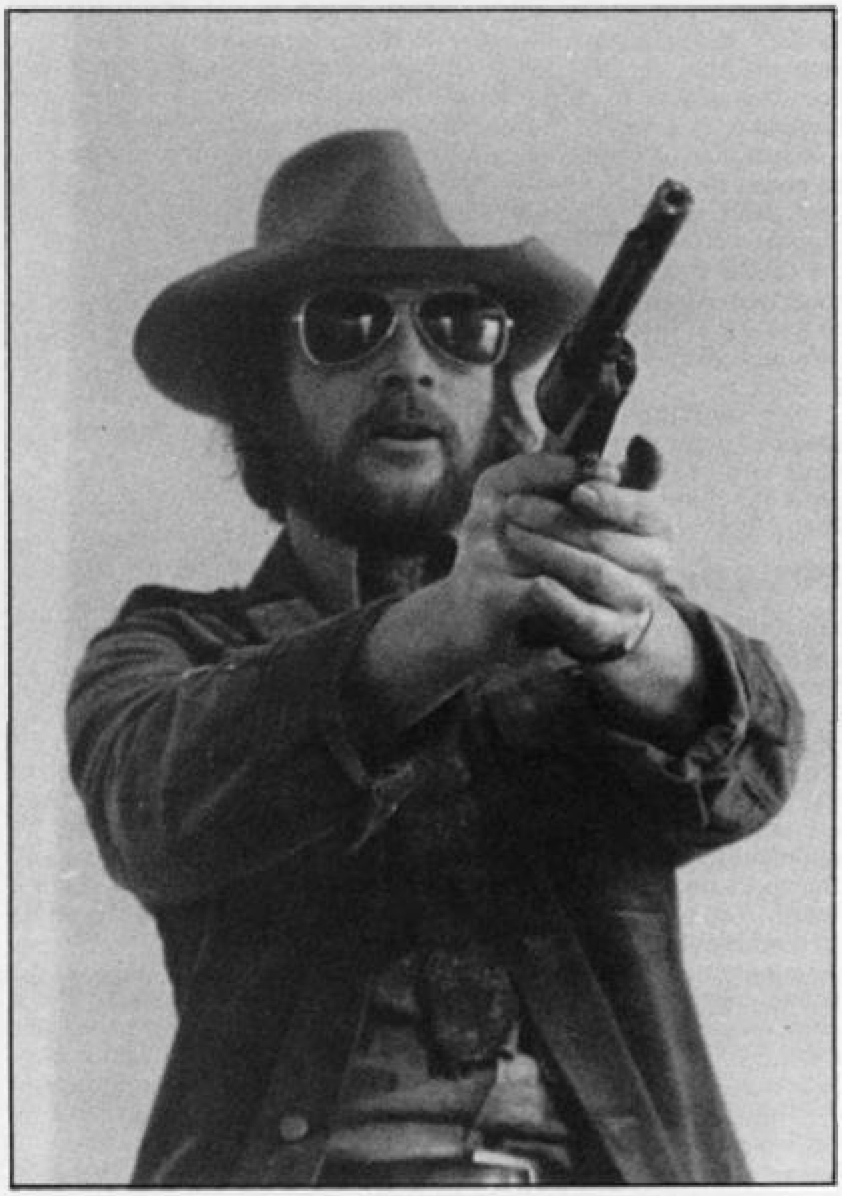
It is very simple to define a graphic. I'll take you through the way I created the alleged duck, and from this explanation you should be able to produce anything you like. The key to the user-defined graphics is an eight by eight grid like that shown below.



To work out your graphic, you simply fill in the squares on this grid which you wish to print as solid dots in the final graphic. Our duck (in grid form) looks like that shown below.



If you look to the lines from 140 onwards in the fourth version of Duck shoot, you'll see a little loop, which includes a READ statement and (line 170) the disturbing-looking POKE USR. . . and so on. There are 21 user-definable graphics on the Spectrum, and you print them by getting into the graphics mode (so the cursor becomes a G) and then pressing the keys from A to U. They will look just like the ordinary letters until they have been re-defined. In fact, when line 40 was first typed in, it was made up from a number of the letter 'A', entered while the computer was in the graphics mode, separated by spaces entered in the ordinary mode.



Once you have a filled-in grid like the one we've just looked at, you have to get the numbers (such as the ones in the DATA statement, line 200) to POKE into the letter so it will come out as a graphic. It is simple to do this. On the bottom row of keys, above the B key, you'll see the word BIN, which stands for 'binary'. Each row across our eight by eight grid can be represented by a single binary number, where an empty square is zero, and a filled-in square is a one. You indicate that you are talking about a binary number by preceding it with the word BIN.

I'll now give you the BIN numbers for each row of squares on the 'duck grid'. Compare them will the filled-in and the empty squares, until you understand how they compare.

BIN 00000000
BIN 00000100
BIN 01001001
BIN 11011110
BIN 00111110
BIN 00001000
BIN 00000000
BIN 00000000

If you look at this pattern of ones and zeros, you'll see that they correspond with the squares on our grid. Now, to convert them to decimal numbers (which are easier to manage) to include in our DATA line (200), we simply type in directly:

PRINT BIN 00000000

for the first line, and a 0 appears. Take note of this zero. Then, try the second number, by typing in directly:

PRINT BIN 00000100

The computer will give the answer 4, since 00000100 (or 100) in binary is 4 in decimal. Note down the 4. Continue to do this, working through each line in turn. This will give you a sequence of eight numbers:

0, 4, 73, 222, 62, 8, 0, 0

You put these in the DATA statement; and put in line 170, within the quote marks, the letter you wish to re-define. Then, by simply RUNning the program, the designated key will be changed. From now on, whenever you press the key 'A' after getting into the graphics mode, the little duck will appear instead. Although the duck will not survive you turning off the computer, it *will* survive a NEW — so you can use graphics defined outside a program within it if you wish to do so. And the graphic will be SAVEd when you SAVE the program. (Note also that there is a change to the end of line 70 in this version of the program with the user-defined duck.)

Try working out a duck-shape of your own on a grid you have drawn up yourself, and enter the numbers for this duck in line 200.

Birds of a feather

The final version of this program we will discuss (Program 5) has three rows of flying ducks. It is best to aim at the middle row of ducks (which fly more quickly than the bottom row) when you RUN this program because they are worth 517 points each, as opposed to the 57 that each of the ducks on the bottom row are worth. The top row is just there to confuse you; the ducks there disappear automatically as the ducks in the middle row are shot, but they cannot be shot directly and they do not contribute to your score.

The middle row of ducks is held in the string, B$, which is set equal to A$ in line 45 as you can see in the listing. Note that you must change the BEEP in line 90. The middle row of ducks is shot at in line 95. If you shoot a duck in the bottom row, your shot ends there — you cannot expect it to continue on to get a duck from the middle row as well! The GOTO 100 at the end of line 90 ensures this.

Line 135 moves the middle and top rows of ducks across, changing the elements in the string by an extra element compared to the changes occuring in line 130. Line 70 prints all three rows of ducks, 'inventing' the top (dummy) row by printing B$ out of register, so the ducks there appear ahead of, although flying in synchronisation with, those in the middle row. This will be clear when you RUN the program.

Room for improvement?

That brings us to the end of this series of Duck shoot games. There are, however, four things you could do to further develop the program:

★ Cut the number of shots available down to make it more challenging.
★ User-define the figure firing the gun, so it is not just an 'X'.
★ Add a 'high score' feature so the game will re-start, preserving a high score you can try and better.
★ Allow the computer to detect when all the ducks have been shot (this will happen when A$ and B$ contain 32 spaces each) and add a bonus to the score if this occurs before all the shots have been fired.

Once you have mastered these simple techniques, try going over some of your earlier programming efforts and 'spice' them up a little. You won't be disappointed . . .

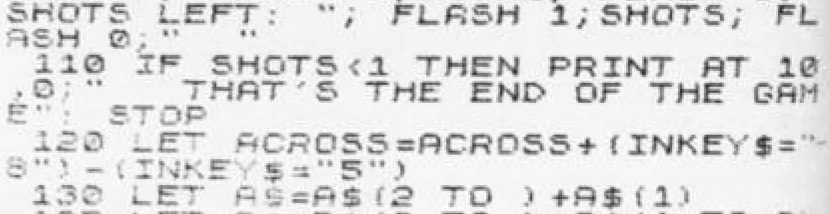
Program 4 — introducing user-definable graphics.

```
  10 REM DUCK SHOOT
  12 GO SUB 150
  15 PAPER 7: BRIGHT 1: CLS : IN
K 2
  20 LET SCORE=0
  25 FOR G=1 TO 20: BEEP .008,G:
 NEXT G
  30 LET SHOTS=15
  35 FOR G=50 TO 20 STEP -2: BEE
P .008,G: NEXT G
  40 LET A$="AAA  A   A  AAAA  AA
AA  A   "
  50 LET ACROSS=15
  60 LET DOWN=14
  70 PRINT AT 7,0; INK RND*6;A$:
 BEEP .008,55-SHOTS
  80 PRINT AT DOWN,ACROSS-1; INK
 2;" X ": BEEP .008,ACROSS
  90 IF INKEY$="F" THEN LET SHOT
S=SHOTS-1: BEEP .05,3*SHOTS: IF
A$(ACROSS)<>" " THEN LET SCORE=S
CORE+57: BEEP .05,50-SHOTS: LET
A$(ACROSS)=" ": BORDER RND*7: BO
RDER RND*7: BORDER RND*7: BORDER
 7
 100 PRINT AT 0,0; INVERSE 1;"SC
ORE: "; FLASH 1;SCORE; FLASH 0;"
SHOTS LEFT: "; FLASH 1;SHOTS; FL
ASH 0;"  "
 110 IF SHOTS<1 THEN PRINT AT 10
,0;"    THAT'S THE END OF THE GAM
E": STOP
 120 LET ACROSS=ACROSS+(INKEY$="
8")-(INKEY$="5")
 130 LET A$=A$(2 TO )+A$(1)
 140 GO TO 70
 150 FOR A=0 TO 6
 160 READ B
 170 POKE USR "A"+A,B
 180 NEXT A
 190 RETURN
 200 DATA 0,4,73,222,62,8,0,0
```

Program 5 — not one, not two, but three lines of ducks for you to shoot at!

```
  10 REM DUCK SHOOT
  12 GO SUB 150
  15 PAPER 7: BRIGHT 1: CLS : IN
K 2
  20 LET SCORE=0
  25 FOR G=1 TO 20: BEEP .008,G:
 NEXT G
  30 LET SHOTS=50
  35 FOR G=50 TO 20 STEP -2: BEE
P .008,G: NEXT G
  40 LET A$="AAA  A   A  AAAA  AA
AA  A   "
  45 LET B$=A$
  50 LET ACROSS=15
  60 LET DOWN=14
  70 PRINT AT 7,0; INK RND*6;A$;
AT 5,0;B$;AT 4,0;B$(3 TO )+B$(1
TO 2): BEEP .008,55-SHOTS
  80 PRINT AT DOWN,ACROSS-1; INK
 2;" X ": BEEP .008,ACROSS
  90 IF INKEY$="F" THEN LET SHOT
S=SHOTS-1: BEEP .05,SHOTS: IF A$
(ACROSS)<>" " THEN LET SCORE=SCO
RE+57: BEEP .05,50-SHOTS: LET A$
(ACROSS)=" ": BORDER RND*7: BORD
ER RND*7: BORDER RND*7: BORDER 7
: GO TO 100
  95 IF INKEY$="F" THEN BEEP .05
,SHOTS: IF B$(ACROSS)<>" " THEN
LET SCORE=SCORE+517: BEEP .05,50
-SHOTS: LET B$(ACROSS)=" ": BORD
ER RND*7: BORDER 7
 100 PRINT AT 0,0; INVERSE 1;"SC
ORE: "; FLASH 1;SCORE; FLASH 0;"
SHOTS LEFT: "; FLASH 1;SHOTS; FL
ASH 0;"  "
 110 IF SHOTS<1 THEN PRINT AT 10
,0;"    THAT'S THE END OF THE GAM
E": STOP
 120 LET ACROSS=ACROSS+(INKEY$="
8")-(INKEY$="5")
 130 LET A$=A$(2 TO )+A$(1)
 135 LET B$=B$(3 TO )+A$(1 TO 2)
 140 GO TO 70
 150 FOR A=0 TO 6
 160 READ B
 170 POKE USR "A"+A,B
 180 NEXT A
 190 RETURN
 200 DATA 0,4,73,222,62,8,0,0
```

A screen illustration showing the lines of ducks and score board of Program 5.



A SPECTRUM CONSOLE THAT EVERYONE CAN AFFORD!

For about the cost of the average games cassette you can now have a truly professional console for your ZX SPECTRUM!

At only £7.50 it's an incredible bargain, and will make your programming or playing so much easier.

How did we do it?

By designing a unique method of self-assembly, and using relatively inexpensive materials!

The result is a console that's tough, light, professionally designed and very highly finished!

For instance: it is made of double-laminated corrugated board with a final laminate of tough, spongeable high gloss vinyl.

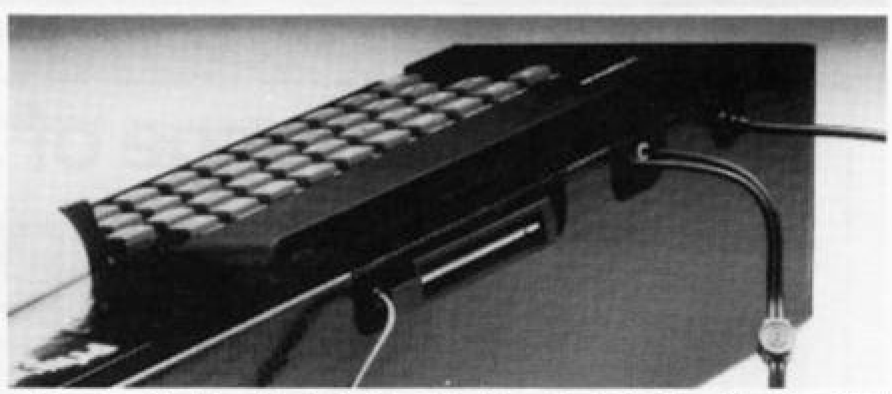
When assembled (about 10-15 minutes) it is completely rigid and tough enough to take years of useful work!

Of course, we designed into it all the facilities you would need for ease of use and performance . . . making it neat, tidy and good looking.

So why pay more for a ZX SPECTRUM console? Send today!







ONLY £7·50

(plus 75p post + packing)

Post today to Print'n'Plotter Products, 19 Borough High Street, London SE1 9SE or phone 01-660 7231 for credit card sales.

Please send me ______ Consoles @ £8.25 (inc Post & Packing)

Please send me illustrated leaflet of Print'n'Plotter products (please enclose stamp).

Remittance enclosed £ ______

Please bill my Access/Barclaycard/Visa/Mastercard No: ______

Name: ______

Address ______

ZA

Please note: Overseas orders please add £1.00 postage for additional surface mail

Print'n'Plotter Products

SEE US AT THE ZX MICROFAIR ALEXANDRA PAVILLION JUNE 4th AND THE COMPUTER FAIR EARLS COURT JUNE 16-19th

Sinclair ZX Spec



The growing range of Spectrum Software



You'll know already that the Spectrum has generated an enormous range of periphera and independent software. O own range is growing very fas and is shown in the Sinclair Software Catalogue – *free* wit every ZX Spectrum.

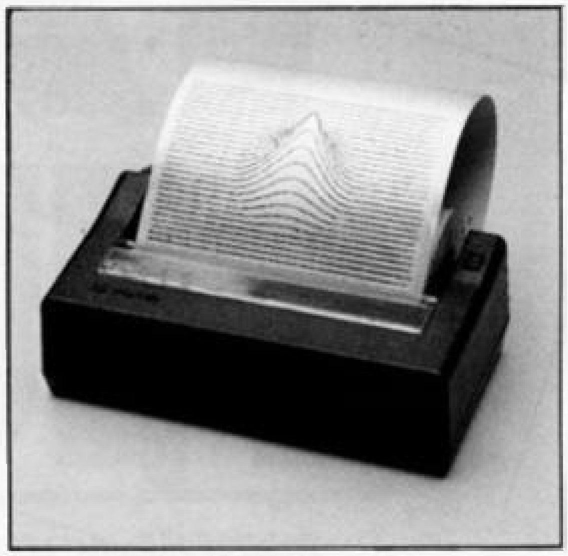
ctrum-news!

16K now £99·95

Previously £125.

48K now £129·95

Previously £175.



ZX Printer now £39.95

Previously £59.95

At last, a 16K colour computer with graphics for under £100!

Why have we done it?

Partly because the sheer volume of Spectrums sold (over 300,000 so far) has brought down unit production costs.

And partly, of course, because we hope you'll buy a Sinclair computer – and not some competitor's promise! We've all heard about colour computers breaking the £100 barrier. Here's the computer that's done it. A colour computer with advanced graphics that's fully supported, and widely available.

Right now, you can order a Sinclair Spectrum at these prices direct from Sinclair on the order form below. And to make it even easier to handle high-level computing at the lowest possible price, we've cut the cost of the printer, too. At £39.95, it's almost unbelievable!

At prices like these, there's really no reason to wait.

How to order your ZX Spectrum

Access, Barclaycard or Trustcard holders – call 01-200 0200 24 hours a day, every day. By FREEPOST – use the coupon below. Please allow up to 28 days for delivery. 14-day money-back option.

sinclair

ZX Spectrum

Sinclair Research Ltd., Stanhope Road, Camberley, Surrey, GU15 3PS.
Tel: 0276 685311. Reg. no: 1135105.

To: Sinclair Research, FREEPOST, Camberley, Surrey, GU15 3BR. **Order**

Qty	Item	Code	Item Price £	Total £
	Sinclair ZX Spectrum – 16K RAM version	3000	99.95	
	Sinclair ZX Spectrum – 48K RAM version	3002	129.95	
	Sinclair ZX Printer	1014	39.95	
	Printer paper (pack of 5 rolls)	1008	11.95	
	Postage and packing: orders under £90	0028	2.95	
	orders over £90	0029	4.95	
			Total £	

Please tick if you require a VAT receipt

*I enclose a cheque/postal order payable to Sinclair Research Ltd for £______

*Please charge to my Access/Barclaycard/Trustcard account no.

*Please delete/complete as applicable.

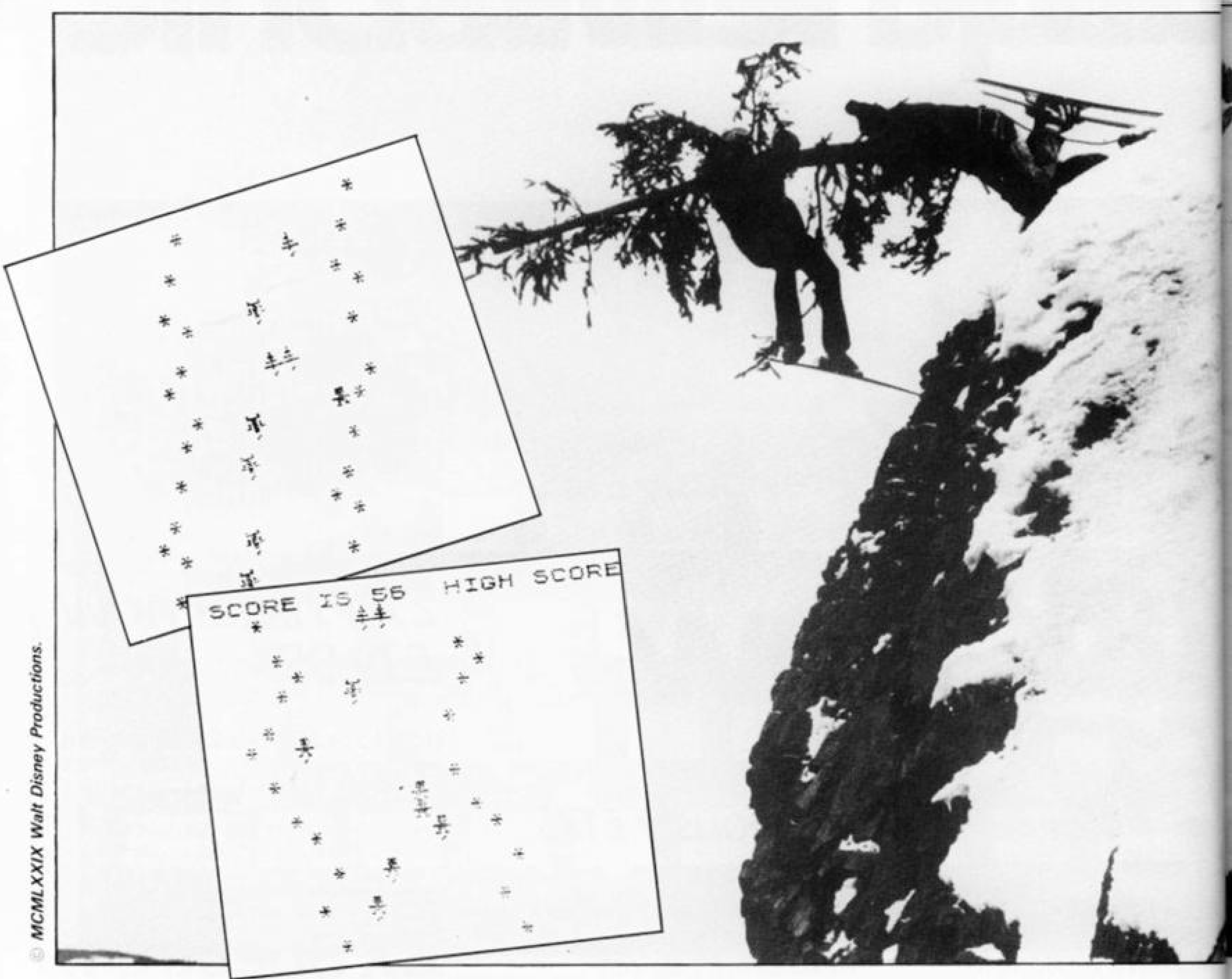
Signature PLEASE PRINT

Name: Mr/Mrs/Miss

Address

ZXC906

FREEPOST – no stamp needed. **Prices apply to UK only.** **Export prices on application.**




```
  1 CLS : BORDER 2
  2 LET h=0
  4 DATA 60,60,60,24,255,60,36,
102
  5 DATA 16,16,56,16,124,16,255
,16
  6 DATA 66,126,24,90,90,219,24
,24
  8 LET u=PEEK 23675+256*PEEK 2
3676
  9 FOR i=0 TO 7
 10 READ j
 11 POKE u+16++i,j
 12 NEXT i
 14 FOR i=0 TO 7
 15 READ j
 16 POKE u+24+i,j
 17 NEXT i
 20 FOR i=0 TO 7
 21 READ j
 22 POKE u+32+i,j
 23 NEXT i
 26 CLS

 27 PRINT INK 1; PAPER 6;AT 2,1
2;"SKI RUN"
 28 PRINT AT 6,2;"Key 'M' steer
s skier to right"
 30 PRINT AT 8,2;"Key 'Z' steer
s skier to left"
 32 PRINT AT 12,4;"For easy gam
e press 'E'"
 33 PRINT AT 14,4;"For hard ga
e press 'H'"
 34 IF INKEY$="e" THEN LET w=10
 35 IF INKEY$="e" THEN GO TO 40
 36 IF INKEY$="h" THEN LET w=7
 37 IF INKEY$="h" THEN GO TO 40
 38 GO TO 27
 40 CLS
 45 LET t=0
 50 LET a=10
 55 LET b=0
 60 LET x=13
 70 LET y=12
 80 LET k=INT (RND*2)
 82 LET l=INT (RND*3)
 90 LET a=a-(k=1 AND a>1)+(k=0
AND a<21)


 95 LET b=b-(l=2 AND b>1)+(l=0
AND b<10)

 110 PRINT AT y,x-1; INK 1;"C":
REM GRAPHICS 'C'

 111 PRINT AT y-1,x-2;"   ";AT y
-2,x-2;"   "

 120 PRINT AT 20,a; INK 4;"*";TA
B a+w;"*"
```



Ski run

Brave the slippery slopes with this program from Mr Davis of Checkley.

If you can't afford that skiing holiday in Switzerland but long for the thrills and spills of the slopes, then this is the program for you and your 16K Spectrum.

Once you've typed it in and got the progam up and RUNning, you'll be greeted with an introduction page explaining the simple rules. (Once you get on the slopes, of course, the going gets a little tougher!) To steer your character down the course, you use the 'm' key to move right and the 'Z' key to move to the left. You also have the option of choosing an easy game ('e') or a hard one ('h').

On the way down the course, you'll find various obstacles in the way of trees. Don't crash into them or you'll find the message 'You have crashed!' come up on the screen and you find yourself back at the beginning. The program also includes a 'High score' facility.

Good luck!

```
 123 IF b=2 THEN PRINT INK 2;AT
20,a+INT (w/2)-2;"D"; INK 7;AT 2
1,a+(w/2)-2;"'" : REM GRAPHICS
'D'

 124 IF b=7 THEN PRINT INK 3;AT
20,a+INT (w/2)-2;"C"; INK 7;AT 2
1,a+(w/2)-2;"'"

 125 IF b=3 THEN PRINT INK 4;AT
20,a+INT (w/2)+1;"D"; INK 0;AT 2
0,a+INT (w/2)+2;"D"; INK 7;AT 21
,a+INT (w/2)+1;"''"

 126 IF b=8 THEN PRINT INK 2;AT
20,a+INT (w/2)+2;"C"; INK 7;AT 2
1,a+INT (w/2)+2;"'"

 127 IF b=5 THEN PRINT INK 0;AT
20,a+INT (w/2)-1;"E"; INK 7;AT 2
1,a+INT (w/2)-1;"'": REM GRAPHIC
S 'E'

 130 PRINT

 140 POKE 23692,-1: PRINT

 150 PRINT INK 0; PAPER 5;AT 0,1
;"SCORE IS ";t;"  "
 155 PRINT INK 0; PAPER 5;AT 0,1
4;"HIGH SCORE IS ";h;"  "

 160 IF SCREEN$ (y,x-1)="*" THEN
 GO TO 205
 165 IF SCREEN$ (y,x-1)="'" THEN
 GO TO 205
 170 LET x=x-(INKEY$="z")+(INKEY
$="m")
 180 LET t=t+1
 190 GO TO 80

 205 CLS
 206 IF t>h THEN LET h=t

 210 PRINT AT 3,5; FLASH 1; BRIG
HT 1;"  You have crashed!!  "

 215 PRINT AT 8,6; FLASH 1; BRIG
HT 1; INK RND*7; PAPER 9;" YOUR
 SCORE IS ";t;"  "

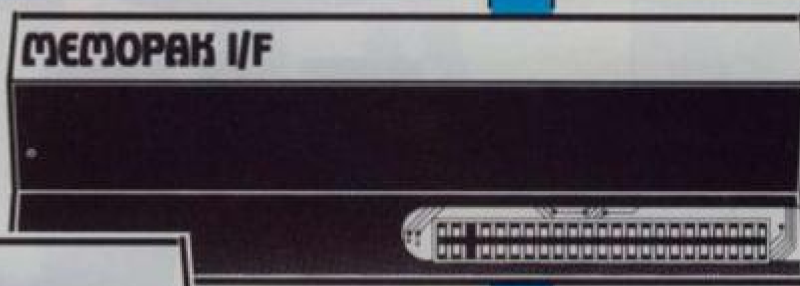
 220 PRINT AT 13,6; FLASH 1; INK
 RND*3; PAPER 9;" HIGH SCORE IS
";h;"  "

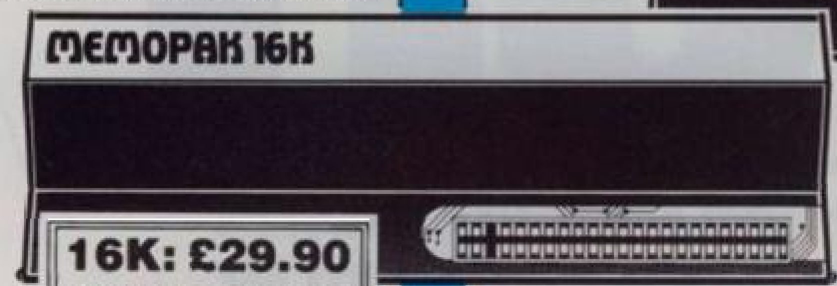
 225 PRINT AT 18,5;"Press 'P' to
 play again"
 230 BEEP .01,RND*20-RND*20

 235 IF INKEY$="p" THEN GO TO 26
 240 GO TO 210
 250 SAVE "ski run"
 260 STOP
```

At Mem…
we realise the potent…

MEMOPAK 16K For those just setting out on the road to real computing, this pack transforms the ZX81 from a toy to a powerful computer. Data storage, extended programming and complex displays become feasible.
For even greater capacity, memory packs can be added together (16+16K or 16+32K).
The MEMOPAK 32K and the MEMOPAK 64K offer large memories at economical prices.





16K: £29.90
32K: £49.95
64K: £79.00
inclusive of VAT

MEMOPAK Centronics I/F
The BASIC commands LPRINT, LLIST and COPY are used to print on any CENTRONICS type printer. All ASCII characters are generated and translation takes place automatically within the pack. Reverse capitals give lower case. Additional facilities allow high resolution printing.
£39.90 incVAT

ZX81

It all adds up to an efficient, modular computer system

The **Memotech** approach to microcomputing is to take the well-proven and popular ZX81 as the heart of a modular system. This small computer houses the powerful Z80A processing unit and acts as the central processor module through which the **Memopaks** operate.

Memotech has a reputation for professional quality, producing units which are designed to fit perfectly, to look well-balanced, and to work efficiently and reliably.

The modular approach gives ZX81 owners the freedom to design the system they really need. Furthermore, the intercompatibility of the modules ensures that later additions will click straight in, to give you a system that grows with your ambitions and abilities.

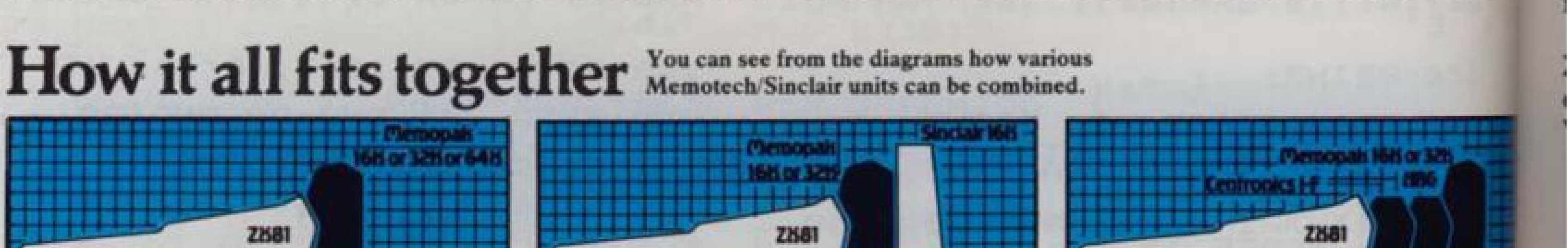
To ensure that your expectations are realised, care is taken at every stage to design features into the system to anticipate your needs. For example:
1) Memories are cumulative e.g. 16K and 32K can be added to the **Memopak 16K** or even to the Sinclair 16K RAM pac… **2)** The **HRG** firmware allows commonly used constructions (such as scrolling, shading and labelling graphs), to be called by a few simple commands. **3)** The **Centronics I/F** converts ZX81 character codes into ASCII and extends the print line… the width of the printer, still using the LLIST, LPRINT and COPY commands.

As one example, a system with 16K of memory and **Memocalc** is all that is required to perform the same sophisticated numerical projections as a computer at 10 times the price. The problem may be as complicated as a cash flow or production schedule, or as simple as household accounts… pocket money budgeting. If your bank manager wants to see… cash flow, then a single print instruction to the **Centronics I/F** will give a printout which is more than acceptable.

The example system which is shown, on the other hand, would satisfy the needs of someone who wanted to enter data…

How it all fits together

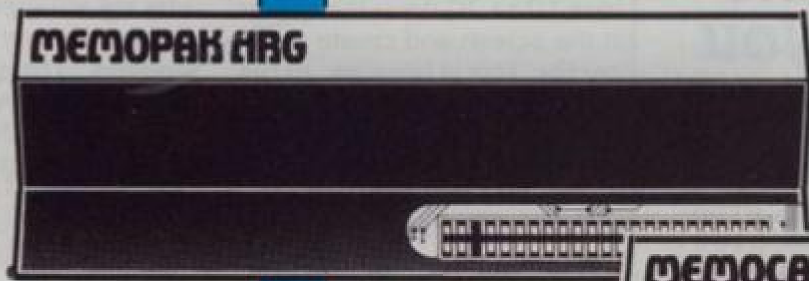
You can see from the diagrams how various Memotech/Sinclair units can be combined.

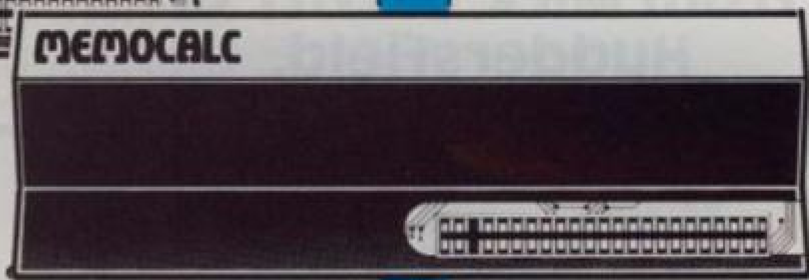


notech, ntial of your ZX81...

MEMOCALC The screen display behaves as a 'window' on a large sheet of paper on which a table of numbers is laid out. The maximum size of the table is determined by the memory capacity, and with a MEMOPAK 64K a table of up to 7000 numbers with up to 250 rows or 99 columns can be specified. Each location in the table can be either a number which is keyed in or a formula which generates a number.

£29.90 inc VAT





MEMOPAK HRG This pack breaks down the constraints imposed by operating at the ZX81 character level and allows high definition displays to be generated. All 248 × 192 individual pixels can be controlled using simple commands, and the built in software enables the user to work interactively at the dot, line, character, block and page levels. Scrolling, flashing and animation are all here.

£39.90 inc VAT

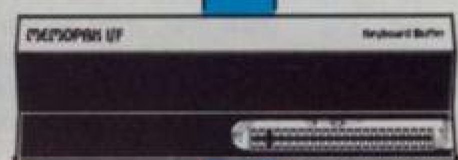


MEMOTECH KEYBOARD

The Memotech plug-in Keyboard plus buffer pack takes the effort out of data entry for ZX81 users. The Keyboard has a light professional touch and is housed in an elegant aluminium case. The simple plug-in system means that you are not obliged to open up your ZX81, use a soldering iron or invalidate your ZX81 warranty.

£49.95 inc VAT





KEYBOARD BUFFER PAK
The Buffer Pak performs a "housekeeping" function for the Keyboard, interfacing directly with the port at the back of your ZX81.

via a light-touch keyboard, construct and label graphs, and then copy the screen to an 80-column printer. Only 16K of memory is shown here but with additional memory, more than one video page can be stored. Up to 7 pages can be displayed in rapid succession to give animated displays.

Looking forward, **Memotech** will continue to back the ZX81 through 1983 with fast storage devices, pressure sensitive electronic drawing boards and more software packs including a **Wordprocessor**, an **RS232 Interface** and a **Z80 Assembler.**

MEMOTECH PRODUCTS ARE AVAILABLE FROM MAJOR BRANCHES OF W.H.SMITH AND JOHN MENZIES

To: Memotech Limited, Witney, Oxon, OX8 6BX
Tel: Witney 2977, Telex 83372 Memtec G
Please send me the following Memotech products:

Please add £2.00 per item to cover post and packing

I enclose a cheque/P.O. for ____________
or please debit my Access/Barclaycard account number

Signature ____________ Date ______

Name and Address ____________

Machine code catch

Take a ZX80 program, add a pinch of machine code and what have you got — a perfect recipe for a 1K ZX81 program from Mick Garfitt of Huddersfield.

The following program was adapted from the ZX80 game of Catch, by I Soutar, on page 103 of the first edition of *ZX Computing*.

The object was to fit the program onto a 1K ZX81, which on the face of it seemed a fairly straighforward task — WRONG! I achieved it only after re-writing the entire program in machine code, and it has taken some months to perfect. It was originally intended solely for my own amusement, but considering the time and effort that went into writing the program, it seems only fair to share it with the rest of the world.

The game is played in the same way as the original. You move by entering '5' to move left, '6' to move down, '7' to move up and '8' to move right. When you have moved, the ZX81 will place a black square on one of the four sides of your position. If you move onto a black square, the game will end and the ZX81 will tell you how many moves you survived. Scoring over 95 is very good, although the computer does not award gold stars!

To enter the program, begin with the following Hex-loader program:

```
1 REM AAAAAAAAAAA
AAAAAAAAAAAAAAAAAAA
AABBBBBBBBBBBBBBBBB
BBBBBBBBBBBBBBBCCCC
CCCCCCCCCCCCCCCCCCC
CCCCCCCCCDDDDDDDDDD
DDDDDDDDDDDDDDDDDDD
DDDEEEEEEEEEEEEEEEE
EEEEEEE
```

(The REM statement must contain 151 characters)

```
10 LET X=16514
20 SCROLL
30 INPUT A$
40 POKE X, 16 *CODE
A$+CODE A$(2)-476
50 PRINT X; " "; A$
60 LET X=X+1
70 GOTO 20
```

When you RUN the above program, the computer will display " " at the bottom of the screen. Enter the first two-digit code shown below, then press Newline. The screen will then show '16514 3E'.

3E is the code you have just entered, and 16514 is the address at which the code is now held. Now enter the remaining codes, pressing Newline after each two-digit code. Enter all the codes in the left-hand column first, then all the codes in the second column, etc.

If you make a mistake when entering the Hex codes, press Newline twice to return to BASIC command, then change X=16514 in line 10 to X=(the value of the address where the mistake occurred). RUN the program again and it will start at the address you have specified without destroying any of the codes entered before that address. When you have entered the last code, press Newline twice to exit from the program. Line 1 should now read REM Y M AND RNDUORNDM, etc. (I always look forward to that bit.) The rest of the program (lines 10-70) is no longer required, and must be deleted and replaced with the following program.

```
10 LET A$="32 inverse spaces"
20 LET B$="shifted graphic 5; 30 spaces; shifted graphic 8"
30 PRINT A$
40 FOR I=1 TO 9
50 PRINT B$
60 NEXT I
70 PRINT A$
80 RAND USR 16514
90 PRINT PEEK 16602
```

Lines 10 to 70 draw the border on the screen and create a display file. This is because, to the best of my knowledge, there is no convenient way of creating a display file using machine code on a 1K ZX81. Line 80 calls the USR subroutine from within the REM statement and line 90 prints the score. Before running this program, ensure that the ZX81 is in Slow mode, or else the screen will go blank. If this should accidentally happen, keep a finger on one of the keys '5' to '8' and the program should return after a few seconds.

Finally, I am including a fully documented copy of my original machine code program, as I believe it would be of immense value to machine code boffins who are struggling, as I was for so long, without making any real progress. The program shows fairly clearly how to print information onto the screen, and how to mask the keyboard for inputs. Those were the two main points which I found confusing, and none of the literature available to me was particularly helpful in this respect.

3E	19	3E	28	OB	14	07	18	DF
00	36	F7	15	11	3E	CB	15	FF
32	14	DB	FE	21	00	07	FE	19
DA	3E	FE	7B	00	3C	CB	80	18
40	00	FE	28	18	32	08	38	01
3A	06	6F	07	03	DA	A8	06	23
34	FF	28	18	11	40	32	11	36
40	0E	12	E6	DF	3E	E0	21	80
32	FF	3E	36	FF	00	40	00	2A
E0	0D	EF	00	36	47	22	19	17
40	B9	DB	2B	00	F6	17	18	41
2A	20	FE	18	19	DF	41	0B	C3
0C	FC	FE	10	7E	EE	FE	FE	96
40	05	6F	36	FE	DF	C0	40	40
11	B8	28	00	00	CB	38	38	00
B5	20	14	23	C0		03	06	00
00	F6	FE	18	36		2B	11	
		77						

The Hex codes you have to enter using the loader program.

Object Code			Source Program	
Decimal Address	Hex Address	Op-Code	Mnemonic	Comments
16514	4082	3E	LDA,00	Initialise variables
16515	4083	00		set score to zero
16516	4084	32	LD(score),A	
16517	4085	DA		
16518	4086	40		
16519	4087	3A	LDA,(FRAMES)	get seed for random
16520	4088	34		number generator
16521	4089	40		
16522	408A	32	LD(seed),A	
16523	408B	E0		
16524	408C	40		
				Plot initial
				position
16525	408D	2A	LD HL,(D-FILE)	find address of
16526	408E	0C		first character on
16527	408F	40		screen
16528	4090	11	LD DE,00B5h	add 181 to that
16529	4091	B5		address
16530	4092	00		
16531	4093	19	ADD HL,DE	
16532	4094	36	LD(HL),14h	print initial
16533	4095	14		position on screen

```
                                          Time delay loop
16534 4096 3E     Delay:LDA,00            set A to zero
16535 4097 00
16536 4098 06     LDB,FFh                 set B to 255
16537 4099 FF
16538 409A OE     Loop 1:LDC,FFh          set C to 255
16539 409B FF
16540 409C OD     Loop 2:DEC C            decrement C
16541 409D B9     CP C                    until zero
16542 409E 20     JRNZ,loop 2
16543 409F FC
16544 40A0 05     DEC B                   decrement B
16545 40A1 B8     CP B                    until zero
16546 40A2 20     JRNZ,loop 1
16547 40A3 F6
                                          Examine keyboard
16548 40A4 3E     Match:LDA,F7h           look at block 3
16549 40A5 F7
16550 40A6 DB     IN A,(FE)
16551 40A7 FE
16552 40A8 FE     CP 6Fh                  look at key '5'
16553 40A9 6F
16554 40AA 28     JRZ,left
16555 40AB 12
16556 40AC 3E     LDA,EFh                 look at block 4
16557 40AD EF
16558 40AE DB     IN A,(FE)
16559 40AF FE
16560 40B0 FE     CP 6Fh                  look at key '6'
16561 40B1 6F
16562 40B2 28     JRZ,down
16563 40B3 14
16564 40B4 FE     CP 77h                  look at key '7'
16565 40B5 77
16566 40B6 28     JRZ,up
16567 40B7 15
16568 40B8 FE     CP 7Bh                  look at key '8'
16569 40B9 7B
16570 40BA 28     JRZ,right
16571 40BB 07
16572 40BC 18     JR match                repeat if no
16573 40BD E6                             key pressed
                                          Plot new position
16574 40BE 36     Left:LD(HL),00          delete old
16575 40BF 00                             position
16576 40C0 2B     DEC HL                  move to new
16577 40C1 18     JR light                position
16578 40C2 10
16579 40C3 36     Right:LD(HL),00         delete old
16580 40C4 00                             position
16581 40C5 23     INC HL                  move to new
16582 40C6 18     JR light                position
16583 40C7 OB
16584 40C8 11     Down:LD DE,             add one line to
                  0021h                   position
16585 40C9 21
16586 40CA 00
16587 40CB 18     JR dup
16588 40CC 03
16589 40CD 11     Up:LD DE,FFDFh          subtract one line
16590 40CE DF                             from position
16591 40CF FF
16592 40D0 36     Dup: LD(HL),00          delete old
16593 40D1 00                             position
16594 40D2 19     ADD HL,DE               move to new position
16595 40D3 7E     light:LDA,(HL)          check if new
16596 40D4 FE     CP 00                   position occupied
16597 40D5 00
16598 40D6 C0     RET NZ                  end game if occupied
16599 40D7 36     LD(HL),14h              plot new position
16600 40D8 14
16601 40D9 3E     LDA, score              get score
16602 40DA 00     score:
16603 40DB 3C     INC A                   increment score
16604 40DC 32     LD(score),A             store new score
16605 40DD DA
16606 40DE 40
                                          Random number
                                          generator
16607 40DF 3E     LDA,seed                get seed
16608 40E0 00     seed:
16609 40E1 47     LDB,A                   generate
16610 40E2 F6     OR DFh                  next 'random'
16611 40E3 DF                             number in
16612 40E4 EE     XOR DFh                 sequence
16613 40E5 DF
16614 40E6 CB     RLC A
16615 40E7 07
16616 40E8 CB     RLC A
16617 40E9 07
16618 40EA CB     RRC B
16619 40EB 08
16620 40EC A8     XOR B
16621 40ED 32     LD(seed),A              store new
16622 40EE E0                             random number
16623 40EF 40
16624 40F0 22     LD(marker),HL           store present
16625 40F1 17                             position on
16626 40F2 41                             screen
                                          Select position
16627 40F3 FE     CP C0h                  if above 192
16628 40F4 C0                             then print before
16629 40F5 38     JRC, below              position
16630 40F6 03
16631 40F7 2B     DEC HL
16632 40F8 18     JR print
16633 40F9 15
16634 40FA FE     below:CP 80h            if above 128
16635 40FB 80                             then print below
16636 40FC 38     JRC, above              position
16637 40FD 06
16638 40FE 11     LD DE,0021h
16639 40FF 21
16640 4100 00
16641 4101 19     ADD HL,DE
16642 4102 18     JR print
16643 4103 OB
16644 4104 FE     above:CP 40h            if above 64
16645 4105 40                             then print above
16646 4106 38     JRC, after              position
16647 4107 06
16648 4108 11     LD DE,FFDFh
16649 4109 DF
16650 410A FF
16651 410B 19     ADD HL,DE
16652 410C 18     JR print
16653 410D 01
16654 410E 23     after:INC HL            otherwise print
16655 410F 36     print:LD(HL)80h         after print position
16656 4110 80                             on screen
16657 4111 2A     LD HL,(marker)          get position on
16658 4112 17                             screen
16659 4113 41
16660 4114 C3     JP delay                repeat program
16661 4115 96
16662 4116 40
16663 4117 00     marker:
16664 4118 00
```

The fully documented listing of the program.

Invaders

Shoot the aliens before they shoot you in this game for your ZX80 written by Andrew Haslem of Walsall.

You begin this game with ten lives, and each time you fail to destroy one of the aliens you forfeit one of your own precious lives.

Once the game is under way, you will be provided with suitable messages to keep you in touch with how many lives you have left. When the game is ended, you will receive a report of how many lives you lost, and how many invaders you managed to shoot.

```
2   LET A=0
3   LET H=0
4   LET J=0
5   LET I=0
20  PRINT "INVADERS"
22  PRINT "YOU HAVE 10
    LIVES"
25  PRINT
30  IF J=10 THEN GO TO
    300
40  IF I=10 THEN GO TO
    400
50  LET Z=RND(2)
60  PRINT "INVADERS
    HAVE FIRED"
70  IF Z=1 THEN PRINT
    "ONE LIFE LOST"
80  IF Z=2 THEN PRINT
    "MISS"
90  IF Z=1 THEN LET
    J=J+I
100 PRINT "DO YOU WANT
    TO..."
110 PRINT "1=FIRE"
120 PRINT "2=REPORT"
130 INPUT S
140 IF S=1 THEN GO TO
    170
150 IF S=2 THEN GO TO
    220
160 STOP
170 LET K=RND(2)
180 PRINT "YOU HAVE ";
190 IF K=1 THEN PRINT
    "HIT"
200 IF K=2 THEN PRINT
    "MISSED"
210 IF K=1 THEN LET
    I=I+1
212 INPUT A$
213 IF A$="S" THEN STOP
214 CLS
215 GO TO 30
220 PRINT "IF SCORES=10
    THEN END"
230 PRINT "INVADERS=";I
240 PRINT "YOUR LIVES=
    ";J
250 GO TO 212
300 PRINT "BOOOM"
310 PRINT "YOU HAVE
    LOST"
320 LET I=I+1
330 PRINT "AGAIN?"
335 INPUT A$
340 IF A$="YES" THEN
    GO TO 4
350 PRINT "GAMES WON"
360 PRINT "YOU=";A
370 PRINT "INVADERS="
    ;H
380 STOP
400 PRINT "BOOOM"
410 PRINT "YOU HAVE
    WON"
420 LET A=A+1
430 GO TO 320
```

Photograph courtesy of Paramount Pictures.



BOOKSHELF

Patrick Cain takes a look at three new titles for you and your ZX Spectrum.

The Art Of Programming the ZX Spectrum — Mike James

'The Art of Programming the ZX Spectrum' is the third in 'The Art of . . . ' and completes the series written for the Sinclair Micro range. By the co-author of the other two excellent accompaniments to the ZX80 and '81, M. James, the Spectrum edition is a pocket size reader discussing the techniques of successful Spectrum BASIC application.

Throughout the series, the approach has been a simple one, examining the capabilities of the machine and suggesting programming techniques that achieve maximum effect.

The reader is required to have passed elementary programming but little further. By making explanations jargon-free and simple to follow and by illustrating each point with practical examples, advanced insight to programming is offered. At no point is the subject likely to be beyond the grasp of most readers. The result, a book that turns straightforward BASIC into really effective programs, and computer users into programmers.

In this book, the same format is closely followed. Chapter two takes to task the area of graphics and explains some of the myriad terms associated with it. Terms that other computer books and publications often bandy about as everyday words but ones that can easily confuse. Following these explanations are applications where 'User-defined graphics' 'Inverse and Over', 'Bright and Flash' might be used to manipulate the screen display. None of these are an end in themselves but without such knowledge it would be difficult to progress. Further, chapter four deals with high resolution, the graphics commands, how to draw circles and ellipses, where high resolution graphics can be used to enhance games and how their use can be a disadvantage.

Breaking from the theory for a time, there is the 'Arrows Game' which is fun to play and highlights much of the topics discussed. By the time you get through chapter six where moving graphics are introduced, you will be becoming aware of how arcade games like 'Squash' and 'Lunar Landing' are constructed; if not, don't worry as a lengthy explanation of each program is provided.

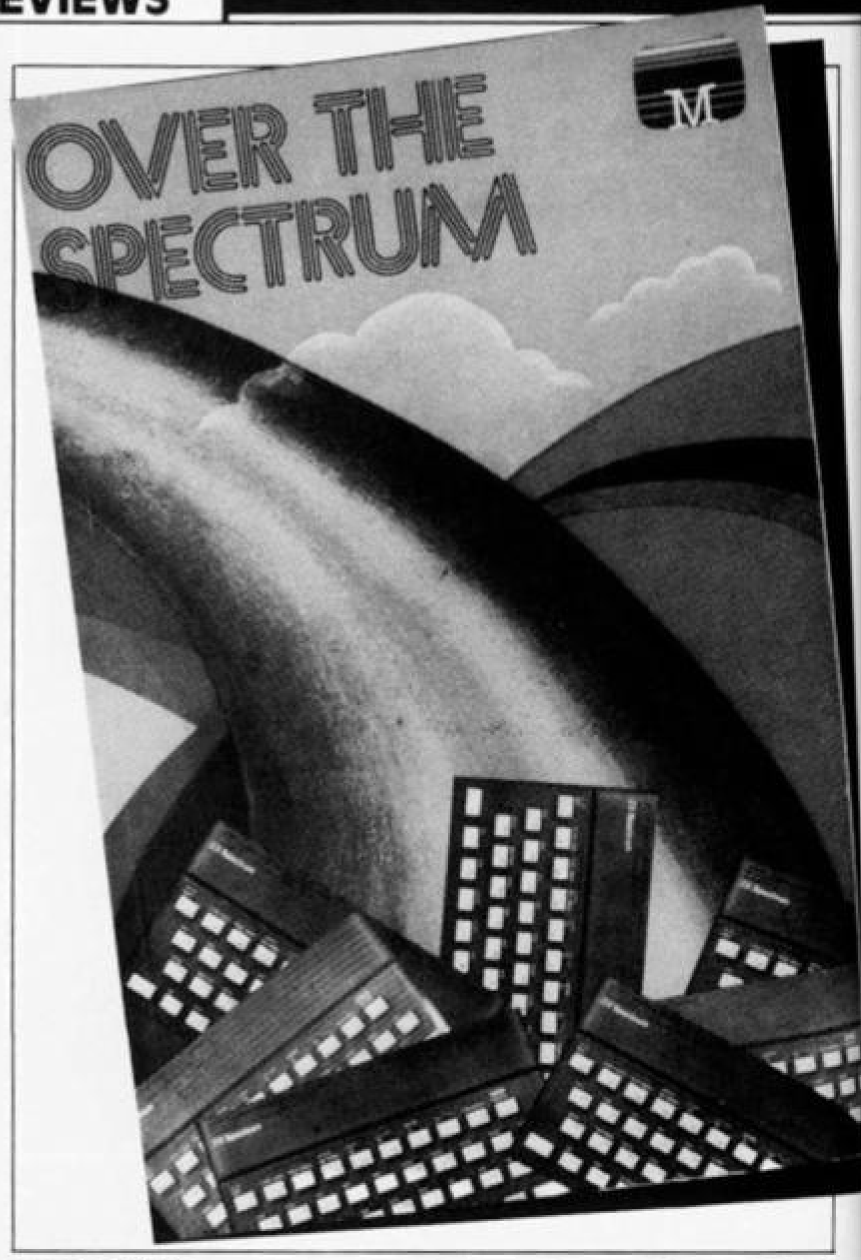
Other programs illustrate more of the techniques of games graphics and by the end of the chapter, you should be in a position to expand and improve upon the programs written and include them in your own games. Indeed, as far as graphics are concerned you ought to be well and truly in the picture!

Chapter five deals with the sound capability of the Spectrum; unfortunately, it is rather limited and anything too complicated is beyond the scope of both the machine and the book.

Maybe no book titled 'The Art of Programming the ZX Spectrum' would be doing its job properly, if it did not investigate PEEK and POKE, as both areas that are fairly advanced; but whilst the bounds of this book do not extend to fully cover the subject, I felt that which was given was a little cursory.

Similarly, areas other than graphics, which gets another airing in chapter ten, were perhaps just too concise, requiring more of the reader and consequently less suited to new readers. Disappointing, as the previous books had usually been suitable for almost all levels of micro buffs.

In comparison to its predecessor 'The Art of Programming the ZX Spectrum' is a poor relation, though in comparison to many books that attempt to do the same sort of thing it is highly recommended and at £2.50, you certainly will not be wasting money. 'The Art of Programming the ZX Spectrum' is written by Mr. James and published in paperback form by Bernard Babani (Publishers) Ltd, The Grampians, Shepherds Bush Road, London W6 7NF.



Over the Spectrum — Melbourne House

'Over the Spectrum' published by Melbourne House, the software people, is a colourful compendium of programs by various authors. And if a computer book of this ilk is to be judged by the games it contains, then the title is neither pretentious or overambitious.

From the outside in, the price of £6.95 is in the upper region of what one might expect to pay for such a book (and perhaps more of that might have been spent on the binding as I found my copy began to fall apart at the seams in its first few days). It does, however, contain eight colour pages. Really? Mine actually had twelve but four of them were double.

Too often I find books of this kind offer little other than poor attempts at reproducing existing arcade games while the Spectrum is suited to running equally exciting and enjoyable games of a different nature. 'Number Reversal' is a game that tests your powers of logic by challenging you to put into order a random series of numbers. The only facility you have to change the sequence of these numbers is the ability to reverse the order of some of the elements. I found it a demanding, and all too often frustrating game, and at only 41 lines long one that could easily be listed and saved.

Longer, but equally impressive, is 'Fruit Machine' — notable for the extensive use made of Spectrum's user-defined graphics capabilities. As the title suggests, the program simulates a fruit machine; since I usually end up losing on the real thing, I found this version to be ideal and just as entertaining with over 60 user defined graphic characters produced on the 16K machine. The listing given in the book does require the 48K RAM; but the accompanying text details how to successfully reduce it to fit aboard the smaller machine.

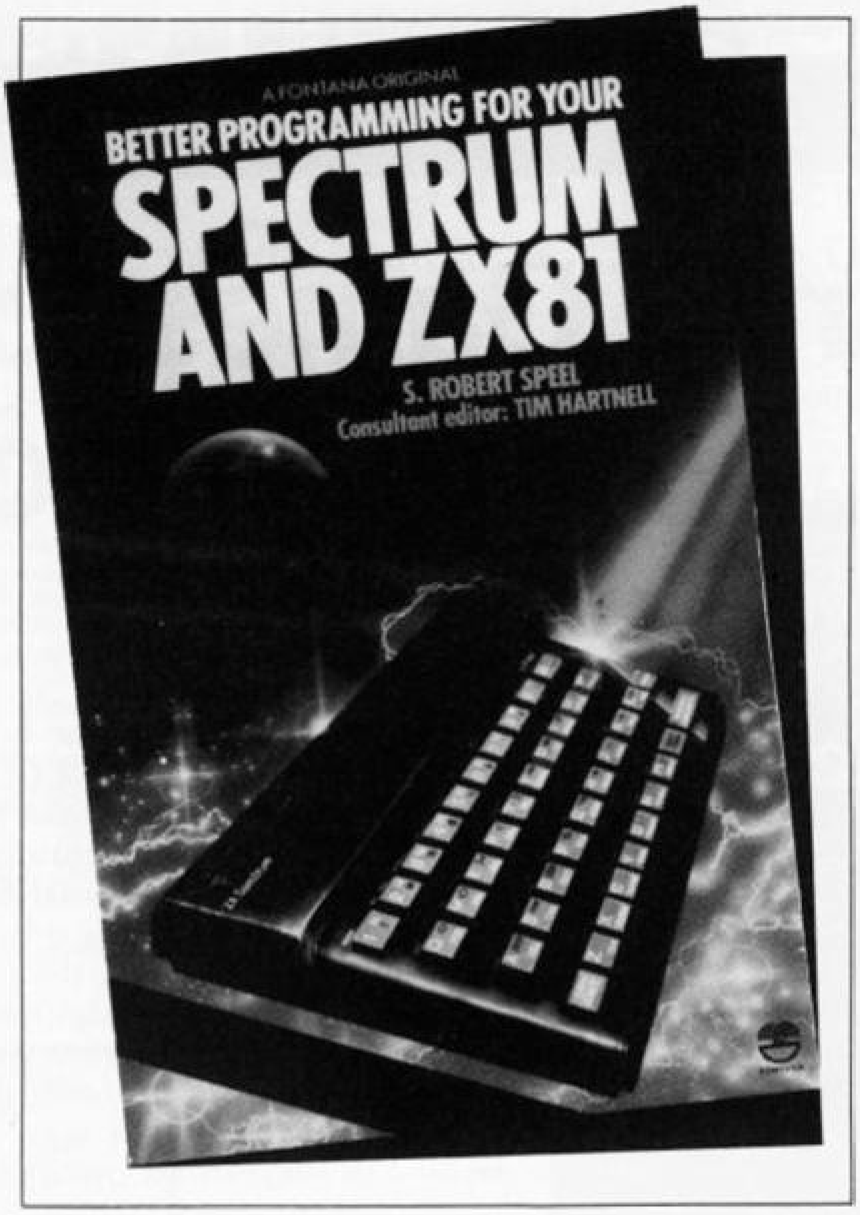
Amongst the rest of those non-arcade type games is a puzzle called 'Leapfrog', 'Blackjack', and '3D Maze Man' in which you are trapped in a maze, escape is against the clock hindered by some unfriendly monsters. What makes

this maze game different from the rest is that by cleverly controlling the graphics, the simulated 3D maze is only revealed in response to your actions on your journey through. The program notes clearly reveal how to alter the appropriate data statements to define your own maze, and consequently the number of possible games is as many as your imagination allows.

Many similar books go to some length to justify the quality of the arcade games they contain, laying blame at the hands of the machine's limitations or those of the language. 'Over the Spectrum' has no need to make such excuses, where the weakness in BASIC exists, machine code has been employed. 'Space Escape', 'Lunar Landing', 'Alien Blitz' and 'Spectrum Invaders' are as good as any commercial versions available and better than most and all are capable of running on the smaller 16K machine. 'Freeway Frog' is particularly notable; by extending the area available for user-defined graphics to allow for 55 rather than the 21 normally available, a great variety of screen characters are produced and their subsequent employment makes for a top quality game.

Of course, games of such a high standard require an equally high standard of programming, often to novice programmers ones that can be difficult to understand. New and experienced programmers can both benefit from the accompanying texts, which offer detailed program descriptions, full program structures and special notes that give advice on adapting the programs to produce others from them. With the novice much in mind, the notes refer the reader to the manual at those points where advanced programming techniques are called upon.

'Over the Spectrum' is a book that you can use the first time you put your fingers to a computer and by listing the programs you will soon be running up to seventeen high quality games. It is also one that will continue to offer advice and instruction as your own skill increases. Published by Melbourne House it is a 164 page collection of programs by a professional software house and costs £6.95. For further information contact Melbourne House, Glebe Cottage, Glebe House, Station Road, Cheddington, Leighton Buzzard, Bedfordshire LU7 7NA.



Better Programming for your Spectrum and ZX81 — Robert Peel

'Better Programming For Your Spectrum and ZX81' as the title proclaims belong to a category of books that I hold some reservations about. It is hardly likely that instructions for maintaining a Rolls Royce will be found side by side in the same volume as those for a Mini. I feel that it is no more likely that one book will serve both the ZX81 and the Spectrum. Has author Robert Peel managed to successfully marry the two tasks in this 284 page publication by Fountain Books?

To find out I flicked through the pages to find the chapter that dealt with converting ZX81 programs to Spectrum ones. It may be true that ZX81 BASIC is essentially a subset of Spectrum BASIC but there are substantial differences in the machines, differences that require lengthy discussion before any effective conversion can take place. The five pages of this section I found were too particular by far to usefully resolve the differences in the two forms of BASIC. It might be said also that of the few programs that would run in the ZX81, there were less than a handful in almost three hundred pages, they were hardly worth converting.

In sizable print, the cover claims over forty new programs are contained; a read down the contents page verifies this as true although a read through the pages that follow shows that good ones number a bit less. There are a few good games, in the main variations on the old faithfulls — 'Sheepdog' and 'Knight Fight', are of this ilk. '3D Maze' is an excellent version of a current favourite; the screen displays the way ahead, and to help when you really get stuck, you can ask for a compass bearing or if lost an ariel view of the maze and your position in it is provided. Several similar games are available currently but this is by far one of the best. Equally deserving of special note is 'Hero Maker' a three part adventure game that can be run on an extended ZX81 or Spectrum.

The program 'Hero Maker' is developed in stages which are added to the core. The game can be run at any point after the core has been entered, a technique that many authors point to as being good programming although all too few however appear to subsequally take any notice of their own advice. Mr. Peel to his credit has, and 'Hero Maker' is a valuable example of how to write a good program as well as being an extremely good game.

I wonder if saying 'I value books that spend some time away from games to look at the educational uses of Sinclair machines' too grand a statement to make? Certainly the educational applications that are possible are extensive, and if you enjoy computing and are studying why not combine the two? I get the feeling Robert Peel might have done just that. His treatment of mainly physics and mathematical problems offer realistic solutions that are indeed valuable for their education qualities rather than their ability to make fancy things happen to the screen. The programs span such subjects as probability, wave properties, quadratic equations and simple trigonometry; the techniques used in their design and construction are explained well and even if none of these subjects are your current headache, the application of similar techniques should soon see you teaching and examining your own educational programs.

Throughout the book, small programs highlight the subjects being discussed — user-defined, high resolution graphics, colour and sound — each supported by a clear if not always detailed accompanying text. They are suited more to Spectrum than the ZX81 but the techniques are general, well-founded, and with little trouble applicable to the smaller machine. The book is not innovative, or obviously suited to a genius, but rather it is aimed at those of us who are learning doggedly by trial and error. Its strength is in that you really sense that the author has recently passed along the same road and has some real advice to offer.

Published by Fontana 'Better Programming For Your Spectrum ZX81' is written by Robert Peel, costs £2.95 and is well worth the cost and time to read it. For further details of this publication you can contact Fontana Paperbacks at 14 St James' Place, London SW1A 1PS.

Club corner



Harlow ITEC

Dear ZX Computing,
Following negotiations with Sinclair Research, Harlow ITEC have solicited support for a Spectrum 48K owners' club. It is proposed that this owners' club shall meet weekly providing a focal point for users, and potential users of Sinclair products. Guest speakers will be arranged from time to time, but generally the format for club meetings will be determined by club members.

For further information on the club and the other activities undertaken by Harlow ITEC, please get in contact with me at the address below.
Yours faithfully,

Rory Waitt,
Senior Training Officer,
Harlow ITEC,
Springhills,
Hobtoe Road,
Harlow,
Essex CM20 1TH.
Tel: Harlow 411048

Royal Air Force Computer Association

Dear ZX Computing,
In early 1982, a few RAF computer enthusiasts floated the idea of forming a Royal Air Force Computer Association (RAFCA). The response was staggering, not only did many individuals come forward, but it was discovered that nearly 40 RAF stations already had computer clubs of their own. There is no doubt that many RAF personnel are keeping well abreast of the latest developments in personal microcomputer technology, and a clear need to assist, support and represent this growing band of enthusiasts is recognised. To this end, an Executive Committee of the embryo RAF Computer Association has been formed and RAFCA is getting 'off the ground'.

The primary objectives of the Association are to encourage and to foster an interest in computing, thereby improving the general level of understanding of applied computer technology to the benefit of the benefit of the individual and the RAF. There is no connection with official Government computing, so the objectives only relate to personal recreational computing. The RAFCA hopes to achieve its aims through presentations, demonstrations, the promotion of competitions and facilities for the exchange of information, ideas and software. The vehicle for much of this information flow will be the RAFCA magazine, 'COMPUTAIR', and its supporting broadsheet, both to be published regularly. RAFCA are seeking advertising support for the magazine, and any offers of advertising, help and discounts from commercial organisations will be gratefully received, and energetically followed-up.

The Association will be launched at the Inaugural General Meeting and Association Day, Nottingham, on Saturday 7 May 1983. It is planned to mount computer and program displays with talks and a celebrity speaker. RAF computer enthusiasts are assured that this will be a worthwhile occasion to attend, and further details will be released nearer the time.

Queries regarding this release should be directed to the Publicity Officer, but more general enquries should be sent to the Secretary, Squadron Leader John Upham RAF, HQ RAFSC, RAF Brampton, Huntingdon, Cambs.
Yours faithfully,

Flight Lieutenant B Otridge RAF

Colchester Sinclair User Group

Dear ZX Computing,
I would be grateful if you could give a bit of publicity to a club that I am endeavouring to set up. The inaugural meeting of the club was scheduled for March 1, and meetings will hopefully happen fortnightly thereafter.
you wish to come and swap ideas and information, is:

Straight Road Centre,
Colchester.

For more information about the club you should 'phone me on Colchester 61066.
Yours faithfully,

Richard Lown,
102 Prettygate Road,
Colchester,
Essex CO3 4EE.

Italian ZX User Club

Dear ZX Computing,
I am writing to tell you about the club we have set up in Italy. Should any of your readers want to write to any of us, you'll find the address below. We also publish a little magazine too.
Yours faithfully,

Arrigo Bondi,
C/O Molino Vecchio 10/F,
40026 Imola,
Italy.

Wandsworth/Paddington Computer Club

Dear ZX Computing,
The Wandsworth Computer Club and the Paddington Computer Club, both recently formed, are now going from strength to strength.

For more information on the Wandsworth Computer Club, contact Howard Cooke at the following address:

West Hill Library,
West Hill,
London SW18.
Tel: 01-874 1144.

For details of the Paddington Computer Club, get in touch with Peter Hill on 01-723 5762.
Yours faithfully,

Robin Bradbeer,
Polytechnic of North London,
London N7 8DB.



The ZX Club

Dear ZX Computing,
On March 22, I was interviewed on Radio Guernsey and although it was only a five minute interview, I received so much response that I decided to start up a club for ZX users. The first meeting of the group took place in April and it is assumed that meetings will occur monthly. The meetings will be held at the following address:

The Old Government House Hotel,
St Annes Place,
St Peter Port,
Guernsey.
Tel: 0481 24921

The club is open to beginners and experts alike — if you own or use a ZX micro, get in touch me soon.
Yours faithfully,

John Lloyd,
Flat 3,
Wightbridge House,
Lapollat,
St Peter Port,
Guernsey.
Tel: 0481 22769

Club Micro Europe/Hobby Computer Club

Dear ZX Computing,
Unfortunately, the BDMA (Belgium-Dutch Microcomputer Club) has now closed. I would be most grateful if you could mention this in your magazine.

However, if you wish to join Micro Europe Club you can contact them at the following addresses:

Club Micro Europe,
Chemin du Moulin 38,
B-1328 Chain,
Belgium.

and:

Hobby Computer Club,
Van Gamerenlei 16,
2130 Braschaat,
Belgium.

Yours faithfully,

P Glenisson,
Priester de L'Epeestratt 14,
1200 Brussels,
Belgium.

Liverpool User Club

Dear ZX Computing,
Our club meets every Wednesday evening between 6.30pm and 9.00pm at The ZX Centre (the address of which is given below).

I am trying to get more people involved in a local group and am at present offering advice on programming and which hardware add-ons to buy for your ZX81 or Spectrum. I also offer some assistance in sorting out problems that people may have with their own programs or those in magazines, both BASIC and machine code.

Incidentally, I would like to hear from anyone with a good knowledge of machine code, as most of the current club members are interested but not really qualified to any great extent in using it or explaining the theory.

For more details, write to my address (see below) or 'phone me on 051-236 6109 during the day.
Yours faithfully,

Keith Archer,
ZX Computing Centre,
17 Sweeting Street,
Liverpool 2.

Camden Computer Club

Dear ZX Computing,
The library of the London Borough of Camden has expressed an interest in the possibility of forming Computer Clubs at its branch libraries. At present, residents of the borough must cross the borders into neighbouring Islington for the computer club at North London Poly, The North London Computer Club. To do this is not necessarily convenient as it is always easier to travel in and out of London's centre than across.

Initially, a computer club will start at the Queens Crescent Library, NW5. The opening date is to be February 8, 1983, at 7.30pm, but meetings are to be scheduled at the same time each week. It is not intended to confine the club to any particular micro, but Sinclair users will, of course, be more than welcome.

Anyone interested in finding out more about the club can contact the library on 01-485 4551, or you can write to me at the address given below or 'phone me on 01-267 1617.
Yours faithfully,

Wil Jackson,
131 Denton,
Malden Crescent,
London NW1.

Wickford Computer Club

Dear ZX Computing,
I am writing to announce a microcomputer club in the Basildon area. We meet fortnightly on a Tuesday evening, 7.30 to 10.00pm, at the following address:

Healey Management Services,
The Hemmells,
Laindon.

For further information, new members should ring me on Wickford 63032 after 6.30pm any night.
Yours faithfully,

Roger Sims,
60 Cedar Avenue,
Wickford,
Essex SS12 9DU.

Eastwood Town Micro Computer Club

Dear ZX Computing,
As a computer club, we have been going for some five months. When we started out, he held one meeting a month; this soon spread to two meetings a week in which we try to cover as much as possible!

January has seen the formation of our new committee and we are looking for new members all the time; at the time of writing we have had 59 enquiries for membership. We usually meet at two venues:

Devonshire Drive Junior School — every Wednesday at 5.45pm.
Eastwood Volunteer Bureau — every Friday at around 6.00pm.

If these are not available, we also meet at Upper School on Mansfield Road or the Sunnycroft Scout HQ on Derby Road.

For further details of the club's activities, contact one of the following:

Ted Ryan — Langley Mill 65011 (Eastwood area).

Roger Hellings — Langley Mill 69281 (Heanor area).

Robert Clifford — Ripley 812459 (Underwood and Selston areas).
Yours faithfully,

Ted Ryan,
15 Queens Square,
Eastwood,
Nottingham,
NG16 3BJ.



A couple of the magazines published by the Italian ZX User Club.



ZX-CESIL 2 part two

John Miller presents the second part of this implementation of the educational computer language, CESIL.

First of all, an apology to those of you who have been patiently awaiting this issue to type in this program in its entirety. Unfortanately, due to the size of the listing, we were unable to carry the full listing in the April/May issue of *ZX Computing* and were only able to publish lines 10 through to 2370.

In this, the second part of this feature, we can now publish the remaining lines of the program, complete with some screen dumps to illustrate the operation of the program. For full instructions on the use of the program, consult the article, ZX-CESIL – part one, in the April/May issue of *ZX Computing*.

```
2380 LET w$="Illegal label error
.": GO TO error
2385 REM A comment?-If so,RETURN
2390 IF j$(1)=";" THEN LET k$(2
TO )=j$: RETURN
2400 LET cn=0
2410 LET n$=j$( TO 3)
2415 REM See if the instr. does
         exist.
2420 FOR c=1 TO 20
2430 IF n$=c$(c) THEN LET cn=c
2440 NEXT c
2450 IF cn THEN GO TO 2470
2460 LET w$="Instruction doesn't
 exist.": GO TO error
2465 REM Check to see if instr.
         is a CESIL command
2470 IF c(cn)>=1e3 THEN GO TO 24
90
2480 LET w$="Command as statemen
t error.": GO TO error
2490 LET k$(2)=CHR$ cn
2500 LET n=INT ((c(cn)-1000)/100
)
2510 IF n THEN GO TO 2540
2512 REM Instruction should have
         no argument.
2515 REM If there is one then
         output error message.
2520 IF j$(4 TO )="
              " THEN RETURN
2530 LET w$="Illegal use of argu
ment error.": GO TO error
2535 REM Does the <SPACE> exist
         between instr. & arg?
```

```
2540 IF j$(4)=" " THEN GO TO 256
0
2550 LET w$="Space missing error
.": GO TO error
2560 LET j$=j$(5 TO )
2570 IF n<>3 THEN GO TO 2640
2575 REM Instr. is jin/jiz/jum
2580 IF j$(1)="l" THEN GO TO 260
0
2590 LET w$="Label missing error
.": GO TO error
2595 REM Valid label after "l"?
2600 IF NOT j$(2)>="0" AND j$(2)
<="9" AND (j$(3)>="0" AND j$(3)<
="9" OR j$(3)=" ") THEN GO TO 23
80
2610 IF VAL j$(2 TO 3)<1 OR VAL
j$(2 TO 3)>99 THEN GO TO 2380
2620 LET k$(3)=CHR$ VAL j$(2 TO
3)
2630 RETURN
2640 IF n<>1 THEN GO TO 2770
2645 REM Instr. is add/div/loa/
                mul/sub
2647 REM Is the arg. valid?
2650 IF j$(1)="a" OR (j$(1)<="9"
AND j$(1)>="0" OR j$(1)="-") TH
EN GO TO 2670
2660 LET w$="Illegal operand.":
GO TO error
2670 IF j$(1)="a" THEN LET k$(3)
="a": LET j$(1)="0"
2680 LET i$=j$
2690 GO SUB incheck
2700 IF valid THEN GO TO 2720
2710 LET w$="Number out of range
error.": GO TO error
2720 IF i$(1)="-" THEN LET i$=ST
R$ (65536+VAL i$)
2730 IF k$(3)="a" THEN IF VAL i$
>99 OR VAL i$<1 THEN GO TO 2830
2740 LET k$(4)=CHR$ INT (VAL i$/
256)
2750 LET k$(5)=CHR$ INT (VAL i$-
(INT (VAL i$/256)*256))
2760 RETURN
2770 IF n<>5 THEN GO TO 2870
2775 REM Instr. is sto
2777 REM Does "a" prefix exist?
2780 IF j$(1)="a" THEN GO TO 280
0
2790 LET w$="Location missing.":
 GO TO error
2795 REM Is address valid?
2800 LET i$=j$(2 TO )
2810 GO SUB incheck
2820 IF valid THEN GO TO 2840
2830 LET w$="Invalid location.":
 GO TO error
2840 LET k$(3)=CHR$ VAL i$
2850 IF VAL i$>99 OR VAL i$<1 TH
EN GO TO 2830
2860 RETURN
2870 IF n<>4 THEN GO TO 3010
2875 REM Instr. is pri
2877 REM Check for opening quote
         "   "   valid text
         "   "   closing quote
2880 IF j$(1)="""" THEN GO TO 29
00
2890 LET w$="Missing quote error
.": GO TO error
2900 LET n$=j$(2 TO )
2910 GO TO 2930
2920 LET w$="Invalid text error.
": GO TO error
2930 FOR c=1 TO LEN n$
2940 IF n$(c)="""" THEN GO TO 29
90
2950 IF n$(c)>=" " AND n$(c)<="U
" THEN GO TO 2970
2960 GO TO 2920
2970 NEXT c
2980 GO TO 2890
2990 LET k$(3 TO )=n$( TO c-1)+r
$
3000 RETURN
3005 REM You should never reach
        line 3010!
3010 LET w$="Impossible error."
3020 REM ##error
3030 LET w$=w$+r$
3040 GO SUB textprint
3045 REM Syntax is bad flag
3050 LET bad=1
3060 RETURN
3070 REM ##fetchno
3075 REM Find a number-
      i) contents of an address
     ii) a constant
3080 LET i$=p$(cline,3 TO 5)
3090 IF i$(1)=" " THEN GO TO 313
0
3100 LET n=CODE i$(2)*256+CODE i
$(3)
3110 LET z=CODE a$(n,1)*256+CODE
 a$(n,2)
3120 GO TO 3140
3130 LET z=CODE i$(2)*256+CODE i
$(3)
3140 IF z>32767 THEN LET z=z-655
36
3150 RETURN
3160 REM ##overflow
3170 IF ABS acc>32767 THEN GO TO
 3190
3180 GO TO next
3190 LET w$=r$+"Arithmetic overf
low"+r$+r$
3200 GO SUB textprint
3210 GO TO cmode
3220 REM ##incheck
3225 REM Check validity of a
        number held in string
        form as i$-overcomes
        Sinclair's odd VAL
3230 LET valid=1
3240 IF i$="" OR i$="-" THEN LET
 valid=0
3250 FOR n=1 TO LEN i$
3260 IF i$(n)<>"-" AND i$(n)<>"
" THEN IF NOT (i$(n)>="0" AND i$
(n)<="9") THEN LET valid=0
3270 NEXT n
3275 IF i$="
          "( TO LEN i$) THEN L
ET valid=0
3280 IF valid THEN IF ABS VAL i$
>32767 THEN LET valid=0
3290 RETURN
3300 REM ##break
3305 REM Test if user is press-
        ing <SPACE>
3310 IF INKEY$=" " THEN GO TO cm
ode
3320 RETURN
3330 REM ##textprint
3335 REM Output text in w$ to
        printer and/or screen
3340 BORDER mode*2+2: INK 0: PAP
ER 7
3350 IF w$="" THEN RETURN
```



```
3360 IF xpos<>0 THEN PRINT TAB x
pos;: IF mode<>1 THEN LPRINT TAB
 xpos;
3370 PRINT w$;
3380 IF mode<>1 THEN LPRINT w$;
3390 LET xpos=0
3400 RETURN
3410 REM ##input
3420 INPUT LINE i$
3430 GO SUB incheck
3440 IF NOT valid THEN GO TO 342
0
3450 RETURN
3460 REM ##cmode
3470 LET w$="Which mode (1 or 2)
?"+r$
3480 GO SUB textprint
3490 GO SUB input
3500 IF VAL i$<1 OR VAL i$>2 THE
N GO TO 3490
3510 LET mode=VAL i$
3520 LET w$="C?"
3530 GO SUB textprint
3540 INPUT LINE y$
3550 LET w$=y$+r$
3560 GO SUB textprint
3570 IF y$="*" THEN GO TO 3710

3580 IF y$="ter" THEN GO TO 4500
3590 LET cn=0
3595 REM Check if valid command.
3600 FOR c=1 TO 20
3610 IF c$(c)=y$ THEN LET cn=c
3620 NEXT c
3630 LET w$=""
3640 IF NOT cn THEN LET w$="What
?"+r$
3650 IF cn<>0 AND cn<>2 AND cn<>
3 AND cn<>4 AND cn<>13 AND cn<>1
6 THEN LET w$="Statement as comm
and error."+r$
3660 IF w$="" THEN GO TO 3690
3670 GO SUB textprint
3680 GO TO 3520
3690 GO TO l(cn): REM Execute
                       command
3700 GO TO 3520
3710 IF lastln<101 THEN GO TO 37
50
3720 LET w$="No more program roo
m."+r$
3730 GO SUB textprint
3740 GO TO 3870
3745 REM Program entry mode
3750 LET w$="?"
3760 GO SUB textprint
3770 INPUT LINE j$
3780 LET w$=j$+r$
3790 GO SUB textprint
3800 IF j$(1)="%" THEN GO TO 367
0
3810 GO SUB syncheck
3820 IF bad THEN GO TO 3750
3830 LET p$(lastln)=k$
3840 IF lab THEN LET l$(lab)=CHR
$ lastln
3850 LET lastln=lastln+1
3860 GO TO 3710
3865 REM Data entry mode
3870 LET w$="Data?"+r$
3880 GO SUB textprint
3890 INPUT LINE t$
3900 LET w$=t$+r$
3910 GO SUB textprint
3920 LET item=0
3930 LET litm=1
3935 REM Is data valid?
3940 FOR c=1 TO LEN t$
3950 IF t$(c)="*" THEN GO TO 411
0
3960 IF t$(c)<>"," THEN GO TO 40
70
3970 LET i$=t$(litm TO c-1)
3980 GO SUB incheck
3990 IF NOT valid THEN GO TO 408
0
4000 IF item=20 THEN GO TO 4120
4010 LET item=item+1
4020 LET xx=VAL i$
4030 IF xx<0 THEN LET xx=xx+6553
6
4040 LET d$(item,1)=CHR$ INT (xx
/256)
4050 LET d$(item,2)=CHR$ INT (xx
-(256*INT (xx/256)))
4060 LET litm=c+1
4070 NEXT c
4080 LET w$=r$+"Data invalid."+r
$
4090 GO SUB textprint
4100 GO TO 3870
4110 GO TO cmode
4120 LET w$=r$+"First 20 items o
nly accepted."+r$
4130 GO SUB textprint
4140 GO TO cmode
4150 REM ##beg
```




```
4160 LET dp=1: REM Reset data
                   pointer
4170 LET acc=0
4175 REM Check for hal instr.
4180 FOR c=1 TO 100
4190 IF p$(c,2)=CHR$ 6 THEN GO T
O 4240
4200 NEXT c
4210 LET w$="No hal instruction.
"+r$
4220 GO SUB textprint
4230 GO TO cmode
4240 LET w$="Results:"+r$+"[redacted]"+r$
4250 GO SUB textprint
4260 LET cline=1
4270 LET cn=CODE p$(cline,2)
4280 IF cn=59 THEN GO TO next
4290 GO TO l(cn): REM Execute
                      line.
4300 IF cline=lastln-1 THEN GO T
O 4340
4310 LET cline=cline+1
4320 GO SUB break: REM Test for
                       (SPACE)

4330 GO TO 4270
4340 LET w$=r$+"Last line reache
d or hal        executed."+r$
4350 GO SUB textprint
4360 GO TO cmode
4370 REM ##labelfn
4375 REM Attempt to find label
        in program.
4380 LET b=CODE p$(cline,3)
4390 IF l$(b)<>CHR$ 0 THEN GO TO
 4430
```

```
4400 LET w$="Called label does n
ot exist."+r$
4410 GO SUB textprint
4420 GO TO cmode
4430 LET z=CODE l$(b)
4440 RETURN
4450 REM ####################
4460 BORDER 7: INK 0: PAPER 7: C
LS
4470 PRINT AT 11,11; INK 2;"ZX-"
; INK 0;"CESIL"; INK 1;"2"
4480 PRINT AT 13,11; BRIGHT 1;"©
J.Miller"
4490 GO TO 10
4500 BORDER 7: PAPER 7: INK 0: C
LS
4510 PRINT "CESIL terminated,tha
nk you for"''"using this program
."
4520 PRINT '''"You are now back
in Spectrum"'''"BASIC command mod
e."
4530 PRINT '''"To re-run program
,enter "; INK 2;"RUN 4460"; INK
0;'"otherwise proceed as require
d!"
4540 STOP : REM That's all folks
```

```
              ZX-CESIL2

This program allows the user to

speak to the computer in CESIL.

The version of CESIL used is

ZX-CESIL2.4 & is unique to the

Sinclair ZX Spectrum with 16K

BASIC in ROM and 48K RAM.

            Press any key
The commands supported are:
 beg,cha,del,lis,new,ter,*(start
program entry.'ter' returns you
to BASIC)

The statements supported are:
 add,div,hal,in,jin,jiz,jum,kin,
 lin,loa,mul,out,pri,sto,sub,;
(comment marker)

See text for explanation of
all these & rules to be obeyed.

All numbers are integers between
-32767 and +32767 (inclusive).

Store locations are a1 to a99 &
labels are l1 to l99.

100 CESIL lines may be stored.

            Press any key
```

The first two pages of introductory text providing a brief explanation of the various commands and statements supported by the program.

```
C?lis
l1   add   100
     out
     lin
     sto  a99
     sub   500
     sto  a1
     jiz  l2
     out
     lin
     loa  a99
     jum  l1
l2   loa  a1
     pri  "location a1="
     out
     lin
     pri  "Enter a number"
     kin
     lin
     pri  "Your number/5 ="
     div   5
     out
     lin
     pri  "This number times 8="
     mul   8
     out
     jin  l3
     pri  "and is positive"
     jum  l99
l3   pri  "and is negative"
l99  lin
 ;now output data string
     in
     out
     mul   -1
     jin  l99
     hal
1,2,3,4,5,6,7,8,9,10,11,12,13,14
,15,16,17,18,-1,*

List complete
Which mode (1 or 2) ?
C?beg
Results:

 100
 -400
 200
 -300
 300
 -200
 400
 -100
 500
location a1= 0
Enter a number
Your number/5 = 5
This number times 8= 40 and is p
ositive
 1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 -1
Last line reached or hal
executed.
Which mode (1 or 2) ?
```

A sample RUN of the program in operation.

```
CESIL terminated,thank you for

using this program.

You are now back in Spectrum

BASIC command mode.


To re-run program,enter RUN 4460
otherwise proceed as required!
```

When you wish to terminate operation of the program, you are provided with the above page of text showing you how to get back into CESIL mode should you wish.

The '81 soft selection

Nick Pearce levels his critical eye at some of the latest software packages for the ZX81.

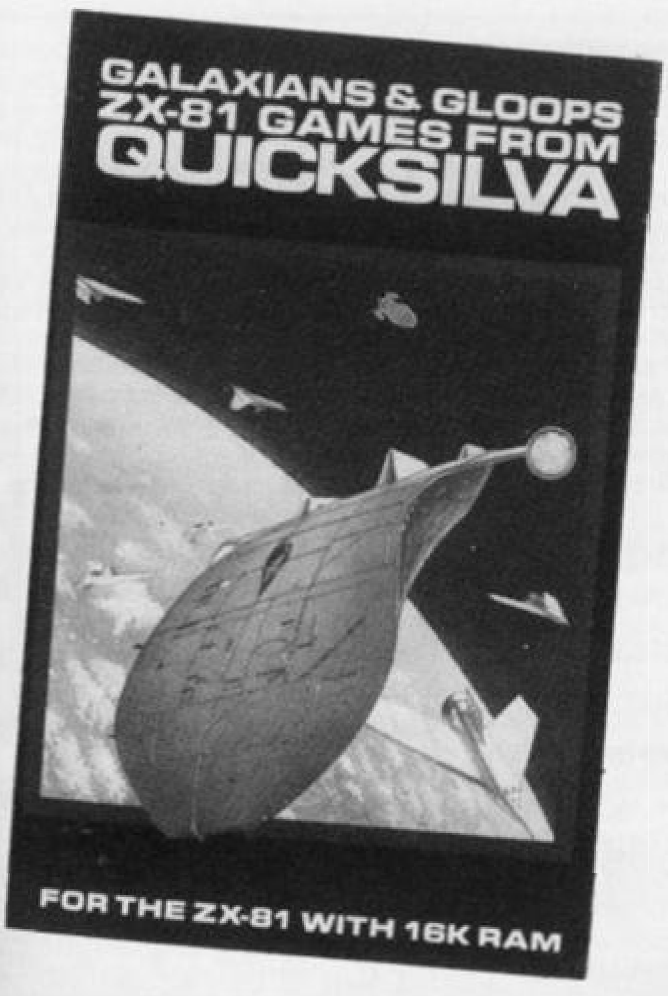
Galaxians and Gloops – Quicksilva

Quicksilva have produced an excellent game of Galaxians for the ZX81. The action is fast and responsive and it is a pleasure to play. Two types of Galaxian move in formation across the top of the screen, and swoop down attacking your base. You have three lives and the speed, firing rate and the number of swooping Galaxians are adjustable by the player. You get more points for hitting Galaxians as they swoop down rather than those in formation. Scoring is displayed on the screen, and at the end of each game your score is entered onto a league table which can cater for up to 15 games/players.

The first batch of swooping Galaxians are particularly good at searching out and destroying your base, and some deft manoeuvering is required to avoid their attack. Thereafter the battle is fierce but a high score can be achieved by the experienced player. If it becomes too easy, the level of difficulty can easily be increased.

A fast-moving and impressive machine code game. Nicely boxed and with Gloops – a maze game – on the B side of this cassette, it is reasonable value for money.

Quicksilva have recently taken on the marketing of Pixel games. The Pixel games have been re-packaged by Quicksilva and have been given attractive covers featuring science fiction artwork. Trader and Subspace Striker are two such games.



Trader

Trader is a three-part space adventure in which you travel round a galaxy of six planets trading with the inhabitants. The object of the game is to make a fat profit on the way whilst avoiding the many and varied pitfalls you will encounter. The graphic displays with this game are very good indeed. There is a short interactive-type game within each of the three parts of this adventure, and a few decisions have to be made – the wares to buy and at what price you sell them, for example. But to a large extent this game moves along at a relaxed pace through the various screen displays with only a little input necessary from the player.

One problem is that each of the three parts of the adventure have to be LOADed separately – the variables are held in a secure memory space to be carried forward to the next part to make a continuous game. The first time I played the complete game took about one hour, but altogether some 20 minutes were spent waiting – impatiently – for the next successive part to LOAD. A game in a number of parts to overcome the limitations of even 16K of RAM (which is soon filled by adventures of this sort) is a good idea, but is more suited to a computer system with fast access disc storage. The slow ZX81 cassette based system makes patience an essential prerequisite – perhaps one of the 'fastload' programs on the market could be used with Trader to good effect.

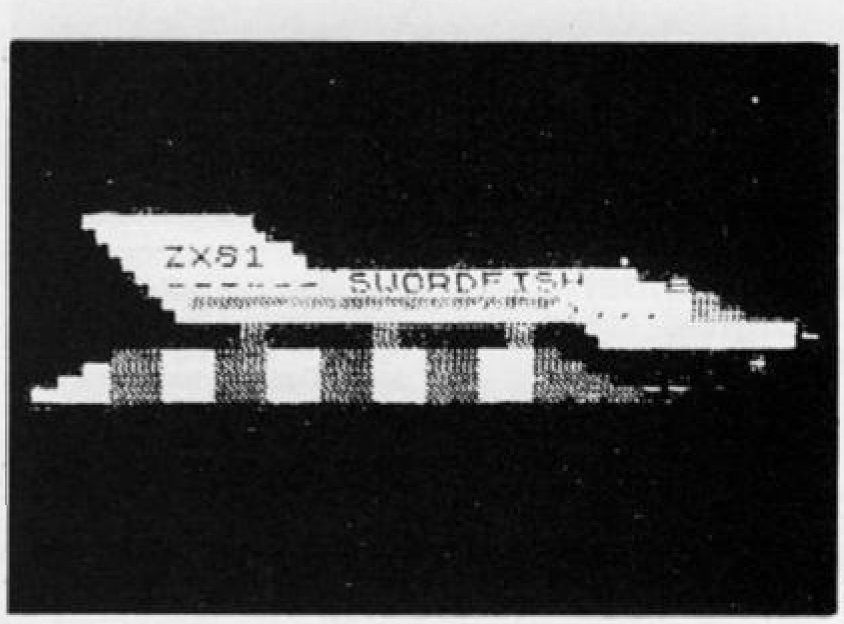
The cassette comes attractively packaged in a box with a comprehensive booklet describing the background to the game and including a Trader story. A good idea, excellent screen displays, and a game to sit back and enjoy. Perhaps a little overpriced at just under £10.

Subspace Striker

Subspace Striker is a more interactive game, but a lot of time is still spent watching the screen displays – too much maybe for the trigger happy arcade game enthusiast itching to shoot down some federation ships with antimat torpedoes. The graphics are certainly very good; each of the four types of ship in the federation fleet, and your own ship – Swordfish – is displayed in great detail on the screen at the appropriate time. The interactive part of the game involves targetting a federation ship in your gun sights and then blasting it from the sky, after which you quickly dive back into the safety of deep space to await the next federation ship to pass. You have ten antimat torpedoes in each game, and there is an element of danger as the federation fleet also has weapons.

Not a game to get the adrenalin really flowing – too much time is spent watching the screen displays for that – but nevertheless an absorbing game which is good fun to play.

The second game on this cassette is Zor, Battle of the Robots. Graphics are a feature of this Pixel game too. In this one, you are on a remote and barren planet and must uphold Earth's honour and supremacy in a battle with a Champion of Zor. You start equally matched with your opponent with 2,500 MJ of energy, which can be used as a weapon or shield. From then it's up to you to use your energy in the best possible way

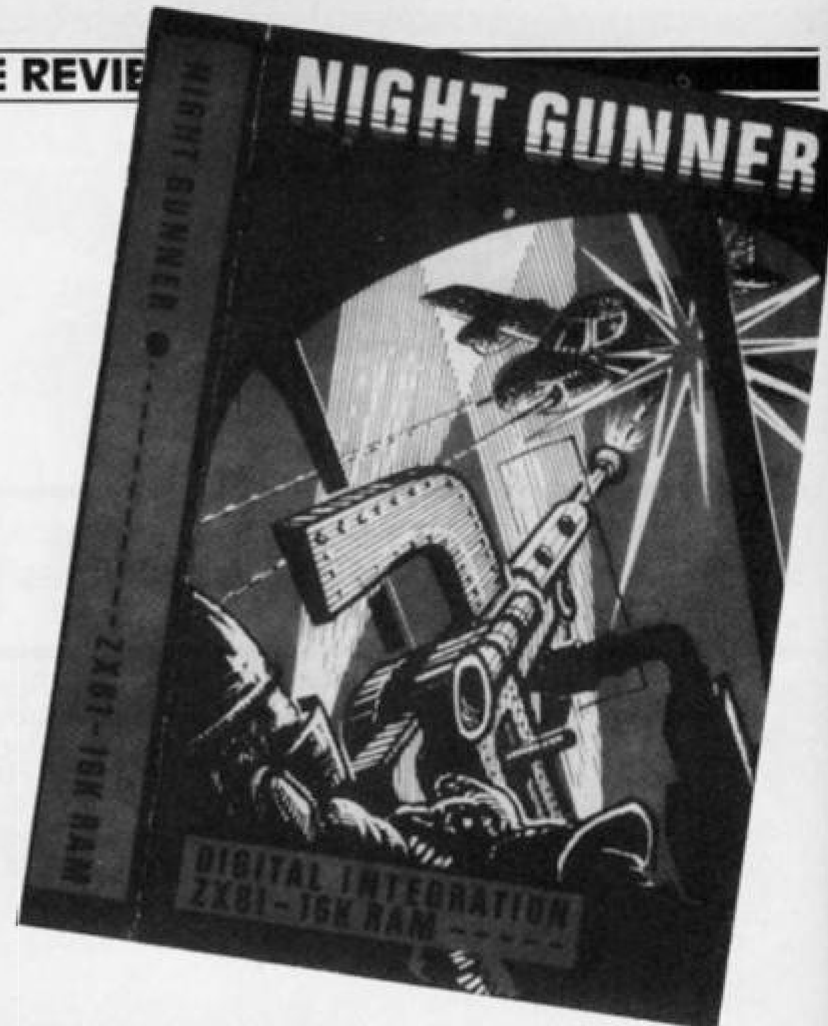


to overcome your adversary – and good luck.

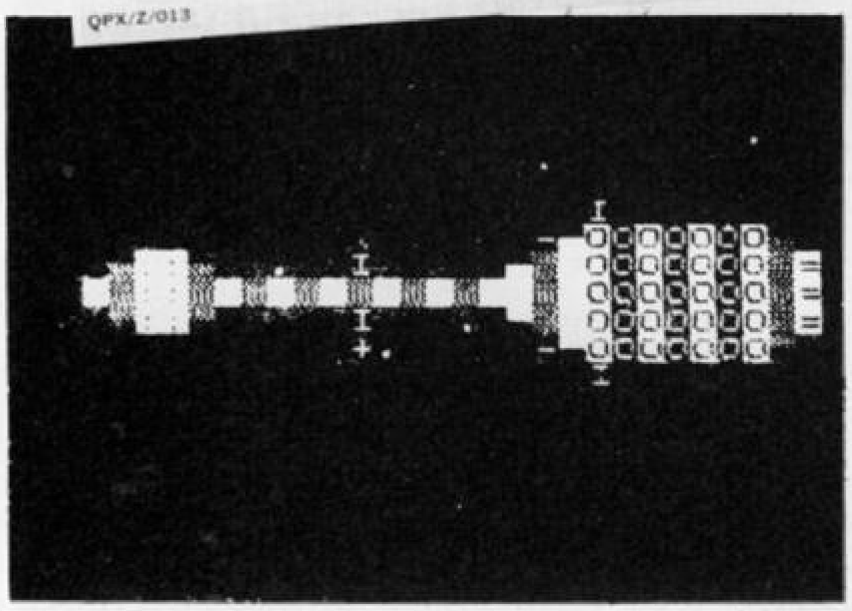
The extensive use of graphics in each game on this cassette means that both take up nearly all 16K of memory, and both take a long time to LOAD. The programs are preceded by a short loading test which can be used to set optimum volume/tone settings on your recorder. I had no problem loading either game.

Quicksilva is at Palmerston Park House, 13 Palmerston Road Southampton. The cassettes reviewed above are priced as follows:

Galaxians + Gloops	*£4.95*
Trader	*£9.95*
Subspace Striker + Zor	*£3.95*







Night Gunner — Digital Integration

In Night Gunner you are the rear gunner in the turret of a bomber aircraft under attack from enemy fighters. The screen display shows the cross wires of your gun sight and a view of the fighters as they weave randomly across the sky. You control your gun using the cursor keys and when you have an enemy fighter in your sights, shoot it down. Don't be too liberal with your ammunition, but you have a limited supply and when it is gone you are a sitting target for the enemy fighters. There are four fighters to shoot down on each mission, and each mission gets successively harder until the game ends with the loss of your third life. Your score is displayed on the screen, and a record of the highest score so far is kept.

Graphics are used to good effect in this game, although I was a little disappointed with the action; for a machine code game it seemed a little unresponsive. Because you control your gun, the enemy fighters appear to move in the opposite direction to your commands. This is a little disconcerting at first, but it makes the game more realistic and it can soon be mastered with some practice. A bit more action during each mission would liven up this game, which nevertheless is challenging and entertaining to play.

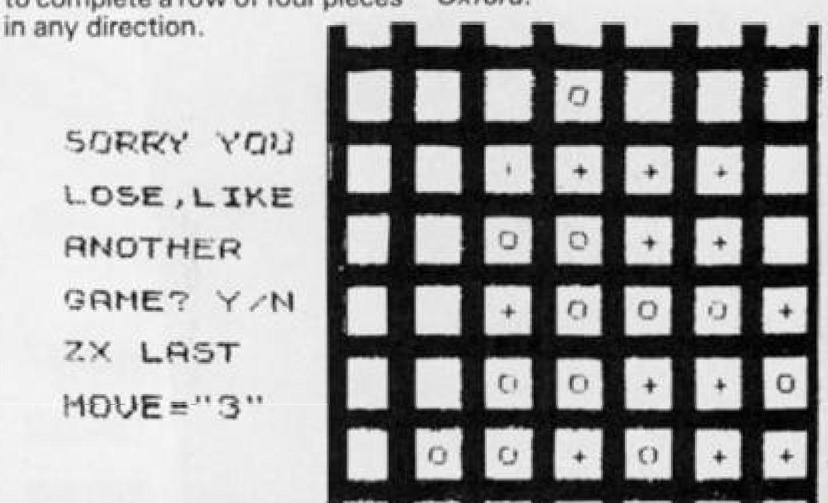
Night Gunner costs £4.95 from Digital Integration, 22 Ash Church Road, Ash, Aldershot, Hants.

ZX Complete Four — Paul Gillett

After all those numbing space battles, ZX Complete Four is something very different — a novel game of strategy to get that grey matter working normally again. Indeed the title is an apt description of the game. You and the computer (or another player) take turns in placing 'pieces' within a frame of seven columns. The winner is the first to complete a row of four pieces in any direction.

The computer doesn't take long to respond to your move, and a game can be completed quite quickly. Unusually, for me, I found I could actually win at this one – against the ZX81 at least. Not much in the way of instructions, which are hardly necessary anyway as you quickly learn the rules whilst you play.

ZX Complete Four costs £3.95 and is available from Paul Gillett, 38 Cromwell Way, Kidlington, Oxford.



Mastering machine code on your Spectrum — part five

Machine code master, Toni Baker, shows you how to incorporate machine code within your BASIC programs with an incredible program to add visual accompaniment to your stereo.

I'd like to cheat a bit now, if I may, by giving you a BASIC program. (Shrieks of horror!) Well, it has got some machine code in it, but nothing new. This is a lesson in how to incorporate machine code into BASIC, for finding a *use* for all these weird and wonderful routines that keep cropping up. Almost the hardest task you have in programming is the hanging around at bus stops and tedious dinner parties waiting for that most ellusive of qualities — *inspiration*.

The machine code routine in question was featured in part three of this series of articles — it's a routine to change the colours of PAPER and INK throughout the whole screen faster than you can say 'The sixth sick sheik's sixth sheep's sick' without falling over. The BASIC that surrounds it is new, however. This is a program to impress your next door neighbours and fanatics of laser shows, or a new way of running a disco. This is visual accompaniment to your stereo!

Sound's great!

It's not technically a sound-to-light unit, for the program has no way of knowing whether or not your stereo is even switched on, let alone what's playing on it. What it is, however, is a very good optical illusion of sound and vision being synchronised. What you do is RUN the program and input answers to the various questions asked, put your favourite record on, switch the lights off and close the curtains, then just sit back while your brain dances round in circles.

The program is featured in Figs. 1 and 2. You should enter the machine code first (using a BASIC program to do so) and then delete this BASIC program to input the one given. Take a look at this now.

I'll now give you some examples on how to RUN the program. Type RUN. (Easy so far, isn't it?) For the question 'NUMBER OF LINES', you should input (say) 5. For 'INK', input 'INT (4*RND) + 4', and for 'PAPER' input '0'. Finally, for 'STARS?' you should input 'Y'. This is just an example — try it out for yourself.

What's the plot?

The program has a couple of extra features which you ought to know about. RUN 200 enables you to define the initial INK and the initial PAPER colours. For instance, RUN 200 and then input 0/7/4/'7 – X'/'7 – Y'/'Y'/('/' counts as ENTER). RUN 400 will SAVE and VERIFY the program and the machine code.

I won't turn the above program into machine code just yet.

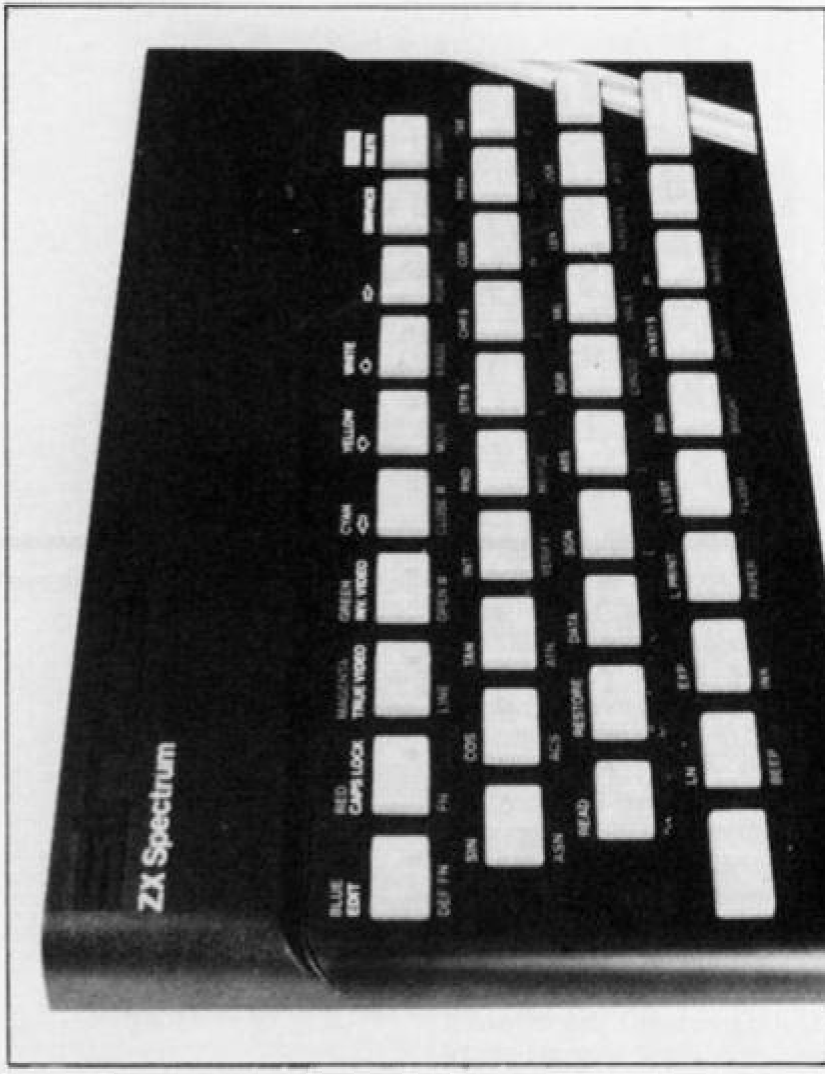
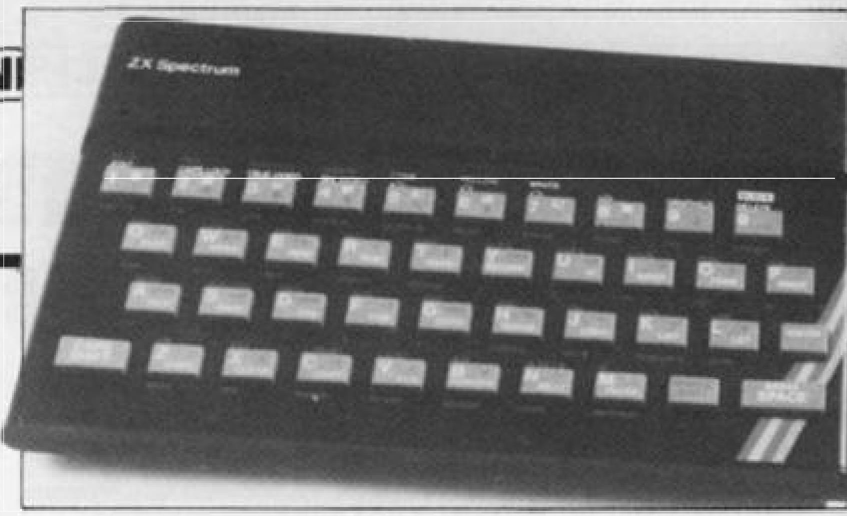


Fig. 1. The program, Patterns; part one — the BASIC.

```
 10 INPUT "NUMBER OF LINES":N
 20 DIM X(N + 1):DIM Y(N + 1)
 30 INPUT " INK ";X$       (use the keyword INK here).
 40 INPUT " PAPER ";Y$       (use the keyword PAPER here).
 50 INPUT "STARS? ";A$: LET S = A$ = "N" OR A$ = "n"
 60 CLS: LET P = 1
 70 LET P1 = P + 1: IF P1 > N + 1 THEN LET P1 = 1
 80 LET P2 = P1 + 1: IF P2 > N + 1 THEN LET P2 = 1
 90 PLOT X(P1),Y(P1)
 95 DRAW X(P2) – X(P1),Y(P2) – Y(P1)
100 PLOT OVER S;X(P1),Y(P1)
105 DRAW OVER 1;X(P2) – X(P1),Y(P2) – Y(P1)
110 LET X(P1) = INT (256*RND)
120 LET Y(P1) = INT (176*RND)
130 PLOT X(P),Y(P)
140 DRAW X(P1) – X(P),Y(P1) – Y(P)
150 LET X = VAL X$: LET Y = VAL Y$
160 POKE 32769,X: POKE 32770,Y: LET P = P1 + USR 32768
170 GO TO 70
200 INPUT "INITIAL INK ";X
210 INPUT "INITIAL PAPER ";Y
220 GO TO 10
400 SAVE "PATTERNS" LINE 430
410 SAVE " PATTERNS" CODE 32768,34
420 VERIFY "": VERIFY "" CODE: STOP
430 LOAD "" CODE: STOP
```



Before I do, I'll give you some information on PLOT and DRAW.

CALL PLOT_BC (Hex CDE522) requires B to hold the Y co-ordinate, and C to hold the X co-ordinate. This will PLOT the required point. Bit zero of P_FLAG (5C91) must be zero for OVER 0 or one for OVER 1.

CALL DRAW_3 (Hex CDBA24) requires B to hold the *absolute value* of the Y parameter, and C the *absolute value* of the X parameter. If Y is greater than or equal to zero then D should hold 01 Hex, otherwise D should hold FF Hex. If X is greater than or equal to zero then E should hold 01 Hex, otherwise E should hold FF Hex. The sequence of instructions LD HL, 2758/EXX *must* be carried out between CAL DRAW_3 and RET (to BASIC) otherwise the Spectrum will crash.

Fig. 2. The program, Patterns; part two — the machine code.

```
010000   PATTERNS  LD BC,???? (To be POKEd by BASIC)
78                 LD A,B
76                 HALT
D3FE               OUT(FE),A
78                 LD A,B
87                 ADD A,A
87                 ADD A,A
87                 ADD A,A
81                 ADD A,C
218D5C             LD HL,ATTR_P
77                 LD (HL),A
210058             LD HL,ATTRS
110158             LD DE,ATTRS+1
01FF02             LD BC,02FF
77                 LD(HL),A
EDBO               LDIR
C9                 RET
```

Transformation time

We still can't rewrite the program into machine code yet, since we don't have a routine for INT (X*RND). It is possible to write a simple random number subroutine which creates random numbers between 0000 and FFFF Hex in the HL register pair, so take a look at Fig. 3 which illustrates such a subroutine. It works using the random number seed already used by the ROM, but does not actually call the ROM's RND routine (since this is highly impractical). The subroutine does, however, work in more or less the same way as RND — it takes the value of the system variable SEED, and multiplies it by some constant. Only the remainder modulo 65536d is taken as the new result. This is not quite RND, but it does give fairly random results.

Fig. 3. The machine code random number subroutine.

```
D5       RAND   PUSH DE
2A765C          LD HL,(SEED)
54              LD D,H
5D              LD E,L
29              ADD HL,HL
29              ADD HL,HL
19              ADD HL,DE
29              ADD HL,HL
29              ADD HL,HL
29              ADD HL,HL
19              ADD HL,DE
22765C          LD (SEED),HL
D1              POP DE
C9              RET
```

Figure Four, on the other hand, is another kettle of fish altogether. This is a subroutine I've called RAND_A in order to distinguish it from the subroutine in Fig. 3. It requires that A contains a number between 00 and A-1. Of particular note is the subroutine MULT which is called from within RND_A; this is a subroutine which will multiply the number held in the A register by the number held in the DE register pair. The result will be formed in the AHL register triplet (the result will always fit in three bytes). RAND_A works just like the ROM's RND routine — it takes a value, SEED, which is multiplied by 4B, and then the remainder is found from a division by Hex 10001, less one. Can you see how the program calculates this remainder? To find the random number required, this new seed is multiplied by A and the high part only becomes the random number.

Now we are almost ready to turn the program Patterns entirely into machine code. Well — not quite entirely as we still can't do VAL. Let's ignore that part for the time being and leave VAL in BASIC. I'd like you now to study Figs. 5 and 6, which list the revised program, now called Patterns 2. If you now compare it with Figs. 1 and 2, see if you can work out how the translation is achieved. Note that the instruction CALL RAND_A is used — this is a reference to the subroutine in Fig. 4.

You can use any addresses you like for this program. I chose to use the following:

Label	Hex	Dec
RAND_A	7000	28672
MULT	7021	28705
FIND_A	7030	28729
PL_DR	703B	28731
INIT	7078	28792
MAIN	7098	28824
P_VAL	7105	28933
P_OVER	7106	28934
P_DRAW	7107	28935
LINES	7108	28936
INK	7109	28937
PAPER	710A	28938
STARS	710B	28939
ARRAY	710C	28940

It is essential, however, that P_OVER and P_DRAW be adjacent, and also that ARRAY points to the start of a segment of spare RAM.

*Fig. 4. The INT(A*RND) routine.*

```
C5        RAND_A  PUSH BC
D5                PUSH DE
E5                PUSH HL
F5                PUSH AF
3E4B              LD A,4B
ED5B765C          LD DE, (SEED)
CD????            CALL MULT
A7                AND A
4F                LD C,A
ED42              SBC, HL,BC
3801              JR C,AA
2B                DEC HL
22765C    AA      LD (SEED),HL
54                LD D,H
5D                LD E,L
F1                POP AF
CD????            CALL MULT
E1                POP HL
D1                POP DE
C1                POP BC
C9                RET
0608      MULT    LD B,08
210000            LD HL,0000
29        LOOP    ADD HL,HL
17                RLA
3003              JR NC,BB
19                ADD HL,DE
```

```
CE00                ADC A,00
10F7      BB        DJNZ LOOP
C9                  RET
```

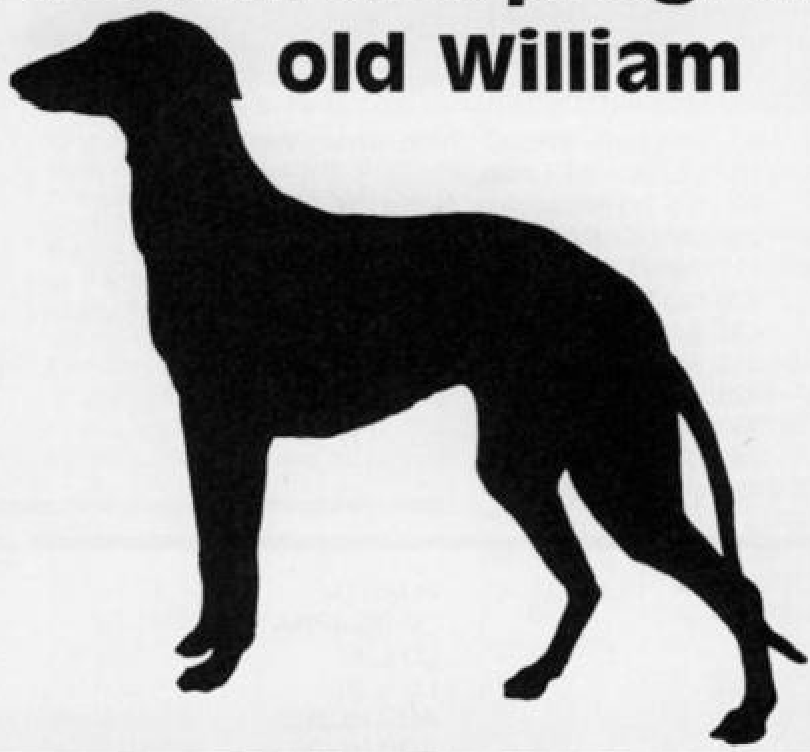
Fig. 5. The program, Patterns 2; the BASIC.

```
10  INPUT "NUMBER OF LINES ";N
20  POKE lines,N: LET N = USR init
30  INPUT " INK ";X$
40  INPUT " PAPER ";Y$
50  INPUT "STARS? ":A$: POKE stars, A$ = "N" OR
    A$ = "n"
60  POKE ink, VAL X$: POKE paper, VAL Y$
70  LET N = USR main
80  GO TO 60
200 INPUT "INITIAL INK ";X
210 INPUT "INITIAL PAPER ":Y
220 GO TO 10
400 SAVE "P2" LINE 450
410 SAVE " P2" CODE rand-a, 263
420 VERIFY ""
430 VERIFY "" CODE
440 STOP
450 LOAD "" CODE
```

```
D5        FIND_A    PUSH DE
11????              LD DE,ARRAY
6F                  LD L,A
2600                LD H,00
29                  ADD HL,HL
19                  ADD HL,DE
D1                  POP DE
C9                  RET
C5        PL_DR     PUSH BC
D5                  PUSH DE
7A                  LD A,D
D5                  PUSH DE
CD????              CALL FIND_A
4E                  LD C,(HL)
23                  INC HL
46                  LD B,(HL)
C5                  PUSH BC
3A????              LD A,(P_OVER)
32915C              LD (P_FLAG),A
CDE522              CALL PLOT_BC
C1                  POP BC
D1                  POP DE
7B                  LD A,E
CD????              CALL FIND_A
7E                  LD A,(HL)
23                  INC HL
1E01                LD E,01
91                  SUB C
3004                JR NC,PD2
ED44                NEG
1EFF                LD E,FF
4F        PD2       LD C,A
7E                  LD A,(HL)
1601                LD D,01
90                  SUB B
3004                JR NC,PD3
ED44                NEG
16FF                LD D,FF
47        PD3       LD B,A
3A????              LD A,(P_DRAW)
32915C              LD (P_FLAG),A
CDBA24              CALL DRAW_3
D1                  POP DE
C1                  POP BC
C9                  RET
210040    INIT      LD HL,D_FILE
110140              LD DE,D_FILE+1
01FF17              LD BC,17FF
3600                LD (HL),00
EDBO                LDIR
ED4B????            LD BC,(LINES-1)
04                  INC B
21????              LD HL,ARRAY
AF                  XOR A
77        IN2       LD (HL),A
23                  INC HL
77                  LD (HL),A
23                  INC HL
10FA                DJNZ IN2
32????              LD (P_VAL),A
C9                  RET
ED4B????  MAIN      LD, BC, (P_VAL)
3A????              LD, A, (LINES)
51                  LD D,C
14                  INC D
BA                  CP D
3002                JR NC,M2
1600                LD D,00
5A        M2        LD E,D
1C                  INC E
BB                  CP E
3002                JR NC,M3
1E00                LD E,00
210000    M3        LD HL,0000
22????              LD (P_OVER),HL
CD????              CALL PL_DR
3A????              LD A,(STARS)
32????              LD (P_OVER),A
3E01                LD A,01
32????              LD (P_DRAW), A
CD????              CALL PL_DR
7A                  LD A,D
CD????              CALL FIND_A
3EFF                LD A,FF
CD????              CALL RAND_A
77                  LD (HL),A
23                  INC HL
3EBO                LD A,BO
CD????              CALL RAND_A
77                  LD (HL),A
5A                  LD E,D
51                  LD D,C
210000              LD HL,0000
22????              LD (P_OVER), HL
CD????              CALL PL_DR
7B                  LD A,E
32????              LD (P_VAL),A
76                  HALT
3A????              LD A,(PAPER)
D3FE                OUT (FE),A
87                  ADD A,A
87                  ADD A,A
87                  ADD A,A
2A????              LD HL,(INK)
B5                  OR L
328F5C              LD (ATTR_T),A
210058              LD HL,ATTRS
110158              LD DE,ATTRS+1
01FF02              LD BC,02FF
77                  LD (HL),A
EDBO                LDIR
215827              LD HL,2758
D9                  EXX
C9                  RET
```

Greyhound racing

All the thrills of the races in this program from fourteen year old William Smith of Stourport.



This program, written for the 16K ZX81, simulates a number of races between four greyhounds. You start the game with £250 and you may bet on any one of the four dogs.

When you first RUN the program, you are welcomed to the races and told how much money you have to bet with. You are then invited to place a bet on one of the dogs.

In the doghouse

Once your bet has been placed, you watch the greyhounds, portrayed as fast moving graphics, race towards the finish line. You are then told which dog won and how much, if anything, you won from your wager.

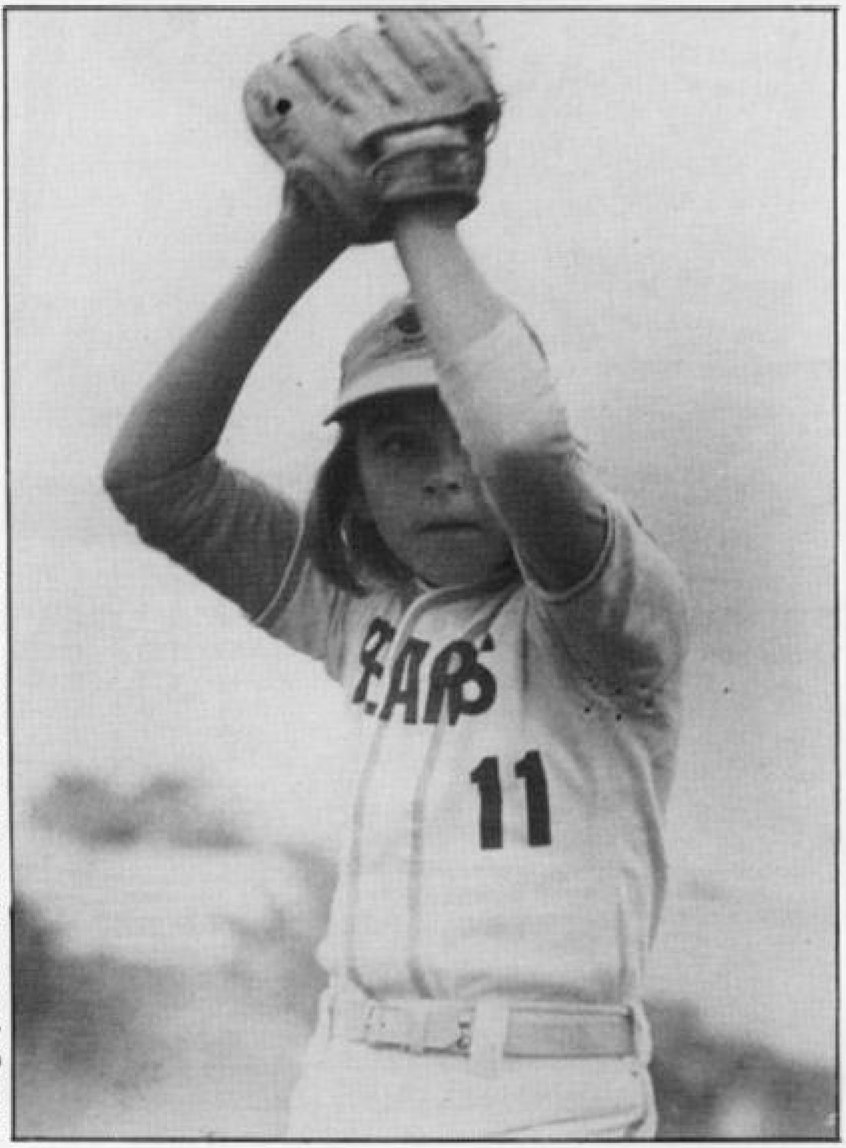
The program then moves on to the next race and once more, you are invited to place another bet. Should you wish to alter the odds, which are initially set at 5-1, you should look closely at line 700.

The game ends when you eventually run out of money to make another bet.

```
GREYHOUND RACES
   1 LET Z=250
  10 LET A=2
  20 LET B=7
  30 LET C=12
  40 LET D=17
  50 LET E=1
  60 LET F=1
  70 LET G=1
  80 LET H=1
 100 LET A$="    "
 110 LET B$="    "
 200 PRINT "WELCOME TO THE GREY
OUND RACES"
 210 PRINT "YOU HAVE ";Z;" POUN
S TO SPEND."
 220 PAUSE 250
 230 CLS
 240 PRINT "ENTER BET 1-";Z
 250 INPUT X
 255 IF X>Z THEN GOTO 250
 260 LET Z=Z-X
 270 PRINT AT 0,0;"           YOUR
ET=";X
 280 PRINT AT 10,10;"WHICH DOG
-4"
 290 INPUT M
 291 FOR N=1 TO 21
 292 PRINT AT N,31;"█"
 293 NEXT N
 300 PRINT AT A,E;A$;AT A+1,E;B
 310 PRINT AT B,F;A$;AT B+1,F;B
 320 PRINT AT C,G;A$;AT C+1,G;B
 330 PRINT AT D,H;A$;AT D+1,H;B
 360 PRINT AT 10,10;"
"
 410 IF E>=25 THEN GOTO 500
 420 IF F>=25 THEN GOTO 550
 430 IF G>=25 THEN GOTO 600
 440 IF H>=25 THEN GOTO 650
 450 LET E=E+(INT (RND*2)+1)
 460 LET F=F+(INT (RND*2)+1)
 470 LET G=G+(INT (RND*2)+1)
 480 LET H=H+(INT (RND*2)+1)
 490 GOTO 300
 500 PAUSE 120
 505 PRINT "DOG 1 FINISHED FIRS
"
 506 PAUSE 120
 507 CLS
 510 IF M=1 THEN GOTO 700
 520 PRINT "BAD LUCK ."
 525 PAUSE 100
 530 CLS
 534 IF Z=0 THEN PRINT "YOU ARE
BANKRUPT"
 536 IF Z=0 THEN STOP
 540 GOTO 2
 550 IF M=2 THEN GOTO 700
 555 PRINT "BAD LUCK."
 560 PRINT "DOG 2 WON "
 563 PAUSE 100
 565 CLS
 570 IF Z=0 THEN PRINT "YOU ARE
BANKRUPT"
 575 IF Z=0 THEN STOP
 580 PAUSE 120
 590 GOTO 2
 600 PRINT "DOG 3 WON"
 602 PAUSE 110
 605 CLS
 610 IF Z=0 THEN PRINT "YOU ARE B
ANKUPT"
 615 IF Z=0 THEN STOP
 620 PAUSE 120
 630 CLS
 635 IF M=3 THEN GOTO 700
 640 GOTO 2
 650 PAUSE 100
 660 CLS
 670 PRINT "DOG 4 WON"
 680 IF M=4 THEN GOTO 700
 700 PRINT AT 10,0;"YOUR DOG WON
";AT 11,0;"YOU WON ";5*X
 710 LET Z=Z+5*X
 720 PAUSE 100
 740 CLS
 750 GOTO 2
```

Every pitcher tells a story

A bat 'n' ball game with an American flavour from Peter Shaw.



Photograph courtesy of Paramount Pictures Corp.

Here's a chance for you to play the American game of Baseball on your 48K Spectrum.

You have a team and the computer has a team, and together you must battle your way through the innings until one team comes out on top. When it is your time to play, one of your team will come up to bat and you will be offered the options of playing safe and hitting down (by pressing the 'd' key) or trying for a home run (by pressing the 'h' key). Once you have made your choice, the screen switches to some great user defined graphics illustrating your batsperson and the pitcher. You'll see the ball fly towards the batsperson and how well your player coped with the pitch.

Once the pitch has been made, the screen reverts back to text and you find out just how well the batsperson did. It could be a home run, or else just a small hit which would place the batter on one of the bases. Of course, if the player was caught out, the next batsperson comes up for a pitch.

As soon as your team have all been struck out, then the computer's team take to the field and your team have to pitch. Once both teams are out, the score is displayed and you either begin another game or start the second innings.

```
  30 PRINT AT 10,10;"Please wait
"
  40 GO SUB 1500: REM UDG's
  50 GO SUB 1430: REM variables
  60 CLS
  70 FOR I=1 TO 9
  80 PRINT "Your innings"
  90 FOR a=1 TO 9: LET w(a)=1: N
EXT a: LET pla=0
 100 FOR m=1 TO 9
 110 IF w(m)=1 THEN GO SUB 450
 120 IF w(m)=0 THEN LET pla=pla+
1: IF pla=3 THEN GO TO 150
 130 NEXT m
 140 GO TO 100
 150 PRINT "Your team are out!"
 160 PRINT "At the end of inning
s ";I
 170 PRINT "your team has scored
 ";hsc
 180 PRINT "The computer now pla
ys!"
 190 LET co=0
 200 INPUT "Press ENTER to conti
nue "; LINE a$
 210 CLS
 220 FOR m=1 TO 9
 230 GO SUB 830
 240 IF co=3 THEN GO TO 250
 250 NEXT m
 260 PRINT "The computer team is
 out !"
 270 INPUT "Press ENTER to conti
nue "; LINE a$: CLS
 280 NEXT I
 290 IF csc>hsc THEN GO TO 330
 300 IF csc=hsc THEN GO TO 350
 310 PRINT "Your team wins ";hsc
;" runs to ";csc
 320 INPUT "Press ENTER to play
again "; LINE a$: GO TO 50
 330 PRINT "The computer wins ";
csc;" to ";hsc
 340 GO TO 320
 350 PRINT "It's a Draw, we must
 have a        second innings!"
 360 FOR a=1 TO 9: LET w(a)=1: N
EXT a: LET pla=0
 370 FOR m=1 TO 9
 380 IF w(m)=1 THEN GO SUB 450
 390 IF w(m)=0 THEN LET pla=pla+
1: IF pla=3 THEN GO TO 250
 400 NEXT m
 410 FOR m=1 TO 9
 420 GO SUB 830
 430 NEXT m
 440 GO TO 290
 450 LET str=1
 460 PRINT "________________________
________"
 470 PRINT P$(m);" comes into ba
t."
 480 LET f=INT (RND*10)+1
 490 PRINT "Will ";p$(m);" hit u
p, and try   for a home run, or
will he hit  down and play safe
? (d or u)"
 500 INPUT LINE h$
 510 CLS
 520 LET ch=((-f)+10)*10
 530 LET an=0: LET an=0: IF h$="
u" OR h$="U" THEN LET an=50
 540 LET ce=(ch*an)/100: LET rn=
INT (RND*25)
```

```
 550 IF rn<ce THEN GO TO 810
 560 IF str=3 AND RND>.9 THEN GO
SUB 1200: LET w(m)=0: RETURN
 570 IF RND>.9 THEN GO SUB 1200:
 LET str=str+1: INPUT "Press ENT
ER for next strike "; LINE a$: G
O TO 540
 580 GO SUB 990: INPUT "Press EN
TER to continue "; LINE a$: CLS
 590 IF f>5 AND h$="u" THEN PRIN
T "The ball is in the outfield,"
 600 IF f<6 AND h$="d" THEN PRIN
T "The ball is in the infield"
 610 PRINT "Your score so far is
 ";hsc: PRINT pla;" player";("s
are" AND pla<>1);(" is" AND pla=
1);" out so far"
 620 LET rn=INT (RND*INT ((f-1))
/1.5)+1: IF rn>4 THEN LET rn=4
 630 IF rn=4 THEN GO TO 650
 640 PRINT p$(m);" steals ";rn;(
"st" AND rn=1);("nd" AND rn=2);(
"rd" AND rn=3);" base"
 650 LET t(m)=rn
 660 IF rn=4 THEN GO TO 760
 670 LET ba=0
 680 FOR a=1 TO m-1
 690 IF t(a)<>4 THEN LET ba=ba+1
: IF ba>2 THEN GO TO 750
 700 IF RND>.5 AND t(a)<>4 THEN
GO TO 730
 710 NEXT a
 720 RETURN
 730 IF RND>.3 THEN GO TO 750
 740 RETURN : PRINT P$(a);" is r
un out.": LET w(a)=0: RETURN
 750 PRINT P$(a);" gets home": L
ET hsc=hsc+1: LET T(a)=4: RETURN

 760 PRINT p$(m);" gets a home r
un !!": LET hsc=hsc+1
 770 FOR c=1 TO m
 780 IF RND>.7 AND t(c)<>4 AND r
n<>4 THEN PRINT p$(c);" is run o
ut": LET t(c)=4: LET w(c)=0: GO
TO 800
 790 IF t(c)<>4 THEN LET t(c)=4:
 PRINT P$(c);" also gets home ":
 LET hsc=hsc+1
 800 NEXT c: RETURN
 810 PRINT P$(m);" is caught": L
ET w(m)=0: LET t(m)=4
 820 RETURN
 830 CLS : LET str=1
 840 LET a=INT (RND*2)
 850 IF a=1 THEN LET h$="d"
 860 IF a=0 THEN LET h$="u"
 870 LET f=INT (RND*10)+1
 880 IF str=3 THEN RETURN
 890 IF RND>.9 THEN GO SUB 1200:
 LET str=str+1: INPUT "Press ENT
ER for next pitch "; LINE a$: GO
 TO 890
 900 PRINT AT 0,0;"________________
________________"
 910 PRINT AT 1,0;"The computer
has ";csc;" runs"
 920 PRINT AT 2,0;"You pitch,
 "

 930 GO SUB 990
 940 PRINT AT 2,10;"and they hit
!"
 950 IF RND>.9 THEN PRINT "They
score a home run !": LET csc=csc
+1: GO TO 970
 960 IF RND>.4 AND a$<>"d" THEN
PRINT "Your team catches the bal
l": LET co=co+1: GO TO 970
 970 INPUT "Press ENTER for next
 pitch "; LINE a$
 980 RETURN
 990 PRINT AT 10,4; INK 6;"GI";A
T 11,4;"CD"
1000 PRINT AT 10,20; INK 5;"EF";
AT 11,20;"CD"
1010 PAUSE 10
1020 PRINT AT 10,4; INK 6;"KL";A
T 11,4;"HJ"
1030 LET di=.1: LET ff=.2
1040 FOR a=45 TO 165 STEP 2
1050 PLOT OVER 1; INK 8;a,95+di
1060 PLOT OVER 1; INK 8;a,95+di
1070 LET di=di+ff
1080 IF a>100 THEN LET ff=ff-.03
1090 NEXT a
1100 PRINT AT 10,20; INK 5;"AB"
1110 IF h$="d" THEN GO TO 1350
1120 LET s=95
1130 FOR a=165 TO 0 STEP -5
1140 PLOT INK 8;a,s
1150 PAUSE 2
1160 PLOT OVER 1; INK 8;a,s
1170 LET s=s+1
1180 NEXT a
1190 RETURN
1200 PRINT AT 10,4; INK 6;"GI";A
T 11,4;"CD"
1210 PRINT AT 10,20; INK 5;"EF";
AT 11,20;"CD"
1220 PAUSE 10
1230 PRINT AT 10,4; INK 6;"KL";A
T 11,4;"HJ"
1240 LET di=.1: LET ff=.2
1250 FOR a=45 TO 190 STEP 2
1260 PLOT OVER 1; INK 8;a,95+di
1270 PAUSE 1
1280 PLOT OVER 1; INK 8;a,95+di
1290 LET di=di+ff
1300 IF a>100 THEN LET ff=ff-.03
1310 NEXT a
1320 PRINT AT 0,0;"Strike ";str
1330 IF str=3 THEN PRINT "Your o
ut !": LET co=co+1
1340 RETURN
1350 LET ff=.3
1360 FOR a=165 TO 30 STEP -3
1370 PLOT INK 8;a,95-ff
1380 PLOT OVER 1; INK 8;a,95-ff
1390 LET ff=ff+.5
1400 IF ff>5 THEN LET ff=-ff
1410 NEXT a
1420 RETURN
1430 BORDER 1: PAPER 1: INK 7: C
LS
1440 DIM w(9): DIM P$(9,8): DIM
c(9): DIM t(9)
1450 LET hsc=0: LET csc=0
1460 RESTORE 1430: FOR a=1 TO 9:
 READ p$(a): NEXT a
1470 LET co=0
1480 DATA "Jenkins","Harwood","Z
eber","Holtop","Gregelis","Todd"
,"Bruda","Pulsman","Fogan"
1490 RETURN
1500 RESTORE 1500: FOR a=USR "a"
 TO USR "l"+7
1510 READ user: POKE a,user
1520 NEXT a: RETURN
1530 DATA 7,7,7,7,7,7,3,3
1540 DATA 0,0,0,224,152,134,128,
128
1550 DATA 7,15,31,25,25,17,32,96
1560 DATA 128,128,128,128,192,24
8,12,4
1570 DATA 3,3,3,3,7,31,97,1
1580 DATA 0,0,192,192,192,192,19
2,192
1590 DATA 12,12,15,7,15,31,63,31
1600 DATA 7,15,14,60,48,48,96,19
2
1610 DATA 0,0,0,192,224,240,240,
224
1620 DATA 224,224,112,48,56,24,1
2,15
1630 DATA 7,7,7,7,15,15,31,31
1640 DATA 0,0,28,252,224,128,128
,128
1650 RETURN
```

Spectrum plotter

Save time and graph paper with this splendid program written by David Elphick of Bristol.

This program has been written for Spectrum owners who frequently make use of user defined graphics within their listings.

The graphic is first plotted eight times the normal size on a grid 16 squares high by 32 squares wide. A cursor (*) is moved around the grid using the direction keys on the Spectrum – the '5', '6', '7' and '8' keys, which will move the cursor left, down, up and right respectively. If you would like to ink in a particular square, simply press Caps Shift and the '5', '6', '7' or '8' key.

Once you have defined the graphic you wish to use in a program (such as that shown in Fig. 1), you simply press the 'G' key to provide the normal size graphic. If this process takes a couple of seconds, don't panic!

If you're not happy with the shape of the graphic, you can change it by pressing the 'C' key and making the necessary alterations. Pressing the 'L' key provides a list of the decimal codes for each of the eight graphics characters. This last key press will provide you with an output something like that shown in Fig. 2.

```
  10 REM User defined graphics p
lotter
  20 REM By David Elphick
  30 DIM a(16,32): DIM b(64)
  50 LET x=0: LET y=0
  60 PRINT AT 20,0;"
                For normal s
ize graphic,press G"
  80 PRINT AT y,x;"*": PLOT 0,11
1: DRAW 255,0: PLOT 64,175: DRAW
 0,-126: PLOT 128,175: DRAW 0,-1
26: PLOT 192,175: DRAW 0,-126: P
RINT AT y,x;" "
 100 LET k=CODE INKEY$
 110 IF k=0 THEN GO TO 80
 120 IF k>7 AND k<12 THEN PRINT
AT y,x;"■": LET a(y+1,x+1)=1
 130 IF INKEY$="g" THEN GO TO 20
0
 135 LET x=x+(k=9)-(k=8)
 140 LET y=y+(k=10)-(k=11)
 145 LET x=x+(k=56)-(k=53)
 150 LET y=y+(k=54)-(k=55)
 160 IF x=-1 THEN LET x=0
 161 IF x=32 THEN LET x=31
 162 IF y=-1 THEN LET y=0
 163 IF y=16 THEN LET y=15
 170 LET a(y+1,x+1)=0
 180 GO TO 80
 200 FOR m=0 TO 3
 210 FOR n=1 TO 16
 220 LET c=16*m+n
 240 LET d=8*m
 250 LET b(c)=a(n,1+d)*128+a(n,2
+d)*64+a(n,3+d)*32+a(n,4+d)*16+a
(n,5+d)*8+a(n,6+d)*4+a(n,7+d)*2+
a(n,8+d)
 260 POKE USR "A"+c-1,b(c)
 280 NEXT n
 290 NEXT m
 300 GO SUB 500
 310 PRINT AT 20,0;"Change patte
rn? Press C.         List codes?-
L.  Run again?-R   "
 320 IF INKEY$="c" THEN GO TO 60
 325 IF INKEY$="l" THEN GO TO 35
0
 330 IF INKEY$="r" THEN RUN
 340 GO TO 320
 350 CLS
 360 PRINT AT 0,0;"A:";AT 8,0;"B
:";AT 0,8;"C:";AT 8,8;"D:";AT 0,
16;"E:";AT 8,16;"F:";AT 0,24;"G:
";AT 8,24;"H:"
 370 FOR m=0 TO 3
 375 FOR n=1 TO 16
 380 PRINT AT n-1,2+8*m;b(n+16*m
)

 385 NEXT n
 390 NEXT m
 400 GO SUB 500
 410 PRINT AT 21,0;"To re-run,pr
ess R"
 420 IF INKEY$="r" THEN RUN
 430 GO TO 420
 500 PRINT AT 17,0;"User graphic
s ACEG = [graphics]";AT 18,14;"BDFH
[graphics]": REM ■ etc = user graphics
 510 RETURN
```

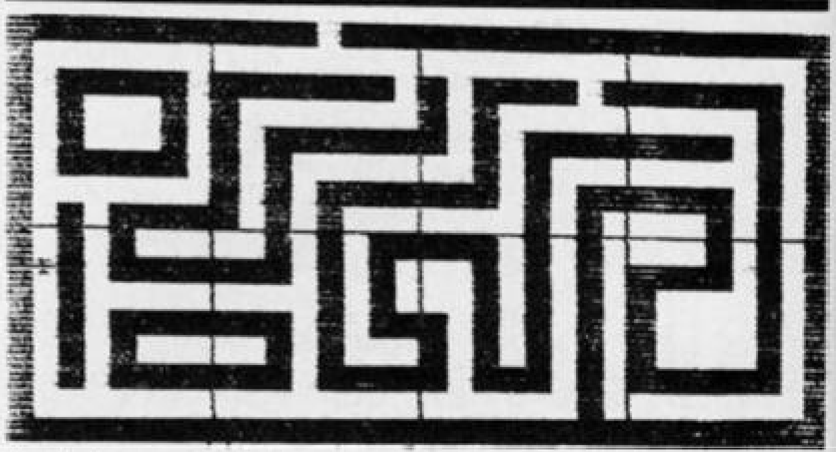


Fig. 1. An example of the kind of user defined graphic that can be obtained with this program.

```
For normal size graphic,press G
A:255    C:255    E:255    G:255
  128      64       0        1
  191      93       247      245
  161      85       20       21
  161      85       215      245
  161      84       80       5
  175      84       87       245
  168      87       212      21
B:168    D:80     F:21     H:245
  175      215      245      1
  160      20       5        125
  175      245      253      69
  160      5        1        69
  191      189      127      125
  128      0        0        1
  255      255      239      255
User graphics ACEG =
              BDFH
```

Fig. 2. The list of decimal codes for the graphic defined in Fig. 1.

Problem page



In this new regular feature, Peter Shaw, programmer extraordinaire, answers a selection of reader's problems.



Peter Shaw, author of 'Games for your Spectrum', will be endeavouring to answer a selection of your programming (and other) problems you encounter on your ZX computer. Should you have any problems, why not send in your queries to the following address:

Problem page,
ZX Computing,
145 Charing Cross Road,
London WC2H 0EE.

Please try and include as much information about your particular problem, including screen dumps, listings, etc.

It must be appreciated that Peter will not be able to answer all of your queiries, but will choose a selection that reflect general problems that should affect you all.

Dear Peter,
When my birthday came I was overjoyed when I was given a ZX81. This was my first computer of my own. Before this I was using the school's Apple computers, which I found hard to program using graphics. I soon started to write some simple programs, which I thought were good for my first attempts, on the ZX81. Then I wanted to add some moving graphics to this, and this is where the problem started, I only knew how to do 'still' graphics by using 'PRINT "(this is where I put my graphics)" '. Could you give me some advice on graphics and how to make them move like space invaders.

Steven Hamilton,
Murdishaw, Cheshire.

Steven,
The problem you have is common amongst beginners on any computer, but once you have found the answer, adding moving graphics becomes simplicity itself. To move your graphics you will have to use 'PRINT TAB', or 'PRINT AT'. The latter of the two is more commonly used because it allows movement all over the screen, whereas 'PRINT TAB' only allows movement along one line. Try this program, and then I will explain how it works:

```
10 LET A = 10
20 LET B = 16
30 PRINT AT A,B; "the same
   number of spaces as there
   are characters in line 70"
40 LET A = A + 1
50 LET B = B + 1
60 IF A > 21 THEN STOP
70 PRINT AT A,B; "your
   graphics"
80 GOTO 30
```

First of all, I defined the variables A and B, then I PRINTed AT A, (10), B, (16) (that is 10 lines down and 16 columns across), the number of spaces required to cover your graphics in line 70, (ie if your graphics were '£/£' then there would be 3 spaces in line 30). I then increased the value of A and B by one and PRINTed your graphics at the new position. Line 60 STOPs the program before the computer tries to PRINT graphics off the screen. Finally, line 80 creates a loop to carry on PRINTing. Try re-reading Chapter 17 of your ZX81 manual and then examine the program above to see if you can understand it any better.

Dear Peter,
Could you please advise me how to get rid of the buzz that emits from Spectrum.

Christopher Looby,
Kilmallock, G. Limerick.

Christopher,
Unfortunately you're stuck with the noise; the only way to get rid of it is to turn the Spectrum off, which is not very practical.

Dear Peter,
I own a 16K Spectrum but now wish to increase the RAM size. There are on the market extensions to RAM which will make the memory 80K. However, I am assured that the Spectrum can only address 48K. Does the larger RAM size present problems and does it mean that part of the extended RAM cannot be used or addressed all the time. Have you any advice to offer or should I simply get a RAM extension to make my Sinclair a 48K machine.

R.J. Weare
Lillington, Leamington Spa.

Mr Weare,
You are right, by the pure nature of the Z80A chip, the Spectrum can only address 48K RAM on top of the 16K ROM. Although I am not familiar with the details of the 80K RAM, most expansions above 48K address in banks, ie while you are using part of the memory, the other part cannot be addressed and vice versa. I recommend that you only go as far as 48K, as software above that will be almost non-existant.

Dear Peter,
I have recently joined the ever growing band of ZX81 users and what I would like to ask you is: out of the wealth of books about the ZX81 and other more general aspects of computing, can you recommend one for me? What I require is as follows: 'a book that will explain how to get the best out of the ZX81, and a book that will explain how some of the add-ons work, and what they exactly do (something that will teach me BASIC in English, if you see what I mean). Lastly, I need a book discussing aspects on how to actually create a functional program.

Vivian Thorpe,
Waddington, Lincolnshire.

Vivian,
There are indeed many books which claim to be the best BASIC programming guides for the ZX81, and it would be difficult to specify just one. Different people learn in different ways. There are two books which I think are particularly good: 'Peek, Poke, Byte & Ram', a witty, but comprehensive guide to BASIC programming, and 'Getting aquainted with your ZX81', an old favourite which teaches you through experience. Luckily enough, there is a book especially for the add-on market, 'The ZX81 Add-On Book', which tries to explain what the add-ons are for. 'The ZX81 Add-On book' and 'Peek, Poke, Byte and Ram' are available from Shiva, 4 Church Lane, Nantwich, Cheshire CW5 5RQ, price £5.50 and £4.95 respectively, and 'Getting aquainted with your ZX81' is available from Interface, 44-45 Earls Court Road, London W8 6EJ, price £5.95.





Dear Peter,
I read the article on 'Programming Skills' in the Dec/Jan issue of *ZX Computing*, and refer to the section 'Press any key to continue'.

Using my 48K Spectrum, I tried to apply the idea to the 'skilful driving' program on page 97 of the same issue, but with rather curious results. I wanted to alter the program so that when none of the cursor keys were depressed, the wheelchair would stop moving. I added:

```
256 GO SUB 720
720 IF IN 61438=255 AND IN
    63486=255 THEN LET
    A=0: LET D=0
730 RETURN
```

This did not work. Upon investigation I found that following the RUN-ENTER command, IN 61438 and IN 63486 always had a value of 191. If the program was broken into and then continued, these ports re-set to 255 (except when a movement key was pressed, of course). The Spectrum manual informed me that the D6 bit was indicating the state of the 'ear' outlet, yet there was nothing connected to this at the time. Could you explain why this should be so?

Also, might other bits of this port (other than bits D0 to D3), be inadvertantly set during a program, (by a sound effect, for example), thus making the idea that 'If the port value is not equal to 255 then a key must be pressed' sometimes unreliable?

John Murray,
Bishopston, Bristol.

John,
This is a problem which I experienced myself while writing a program which required two players to be pushing keys at the same time. This method is indeed unreliable sometimes, but can be overcome by making the first line of your program a BEEP function, ie BEEP .001,0. This is very short, but as you say, it resets the value of the ports to 255, and then your program will work.

Dear Peter,
I have recently purchased a ZX81 computer but have not yet chosen a memory. I am thinking about buying a 16K memory and later possibly adding a 16K slave.

A friend recently loaded a Mikro-Gen cassette game which was accepted by the computer but ran automatically, not being under the control of the Input keys. Yet another friend with a Sinclair memory has no trouble at all.

I also read in Dec/Jan issue of *ZX Computing* the letter from Memotech that the 64K memory was not suitable for the ZX81 yet instructions supplied with the Memopack 16K memory state that this is suitable for use with the ZX81.

(A rather confused)
B. L. Richardson,
Shaw, Oldham.

Mr Richardson,
You did not say whether the cassette LOADed on the Sinclair memory was the Mikro-Gen game, but from a guess i'd say it wasn't. The Mikro-Gen game was probably meant to 'auto-run', that is the game is designed to run automatically so that you can't copy it.

The Memopack 64K expansions will work with the ZX81; the letter was a little confusing, but what it was saying was 'The ZX81 can't normally hold 64K RAM, but our 64K gets around that problem'.

Pilot



Take control of an aeroplane on its last approach to the runway in this program written for us by Bill Longley of Colchester.

MCMLXXX Paramount Pictures Corporation

All you have to do in this game for your ZX Spectrum is to land your 'plane on the runway. However, it's not quite as easy as it sounds!

Your task is to centre yourself East to West, then lower your altitude and cut down the speed of the 'plane. Next, you must put down your undercarriage and, hopefully, land your aircraft. Don't worry if you're not successful at your first few attempts at landing – it often takes many turns before you score any points at all, let alone land the 'plane!

To control your 'plane, you use the '5' and '8' keys to move West and East respectively. A negative number on the control panel means that you are travelling West of the centre of the runway, and a positive number means you are too far over to the East.

In the air tonight

You lower your altitude using the '7' key and you can gain height by pressing the '6' key. To land the aircraft, you must lower your altitude to 100 metres or so, reduce your speed and, when your South reading on the control panel is low enough, land. Your airspeed can be altered using the 'S' key to slow down and the 'F' key to accelerate. Once you have landed, you should use the air brakes to quickly cut your speed – this is done by pressing the 'B' key.

You can only put your undercarriage down when your altitude is less than 1,000 metres. The 'W' or 'D' keys will lower the undercarriage.

It is more important that you should have a low airspeed than to be dead centre of the runway, but you must remember that if your speed goes below 100km/hr then your engines will stall and the inevitable will probably happen.

All of the controls you'll need will be shown to you when you begin the game, and a number of messages appear throughout the game informing you of your status in the air.

The program uses all but 1K of the 16K available and includes quite sophisticated graphics. If you go too far off course you will see some mountains looming onto the screen; should you get too close to the ground, you'll see a forest; and if you complete your mission, you'll get a look at the runway. This last view is almost three-dimensional!



MCMLXXX Paramount Pictures Corporation

```
        _____PILOT_____

     CONTROLS:

     'B' - AIR BRAKES
     'S' - SLOWER
     'F' - FASTER
     'W' - WHEELS DOWN
MOVEMENT:          6 (up)
                   |
                   |
     (west) 5--+--8 (east)
                   |
                   |
                   7 (down)

      HIGH SCORE: 0

     PRESS ANY KEY TO START
```

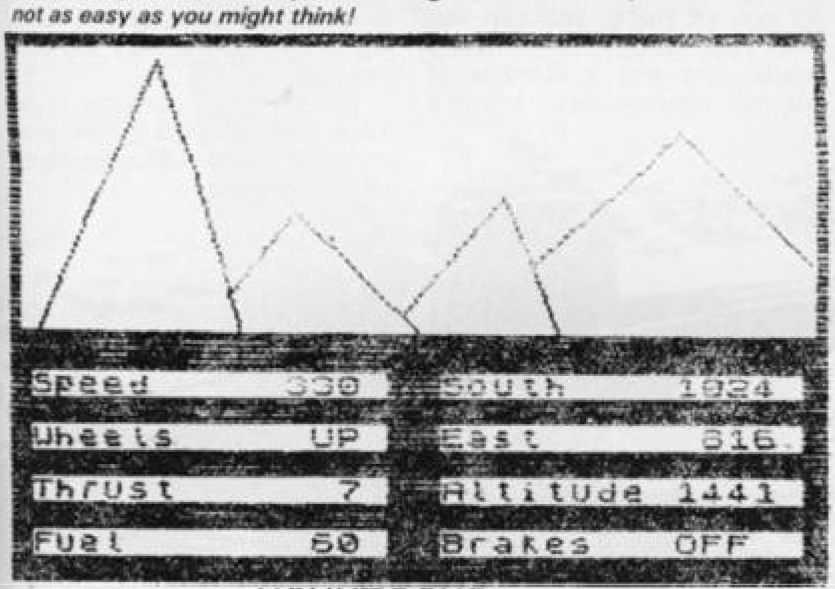
The screenful of rules you are greeted with as you first begin the game.

```
   0 REM *e\ RESTORE MOVE *SIN \CODE GO SUB R RESTORE STR$ <>
   4 LET q=0
   5 LET b$="█♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠█": LET b$=b$+b$: LET b$=b$+b$+b$+b$+b$
   6 GO SUB 8000
  10 CLS : PRINT TAB 8;"_____PILOT_____"
  15 POKE 23658,0: REM SET LOWER CASE MODE
  30 PRINT ''"             CONTROLS:"
  40 PRINT '"          'B' - AIR BRAKES"
  50 PRINT "          'S' - SLOWER"
  60 PRINT "          'F' - FASTER"
  65 PRINT "          'W' - WHEELS DOWN"
  70 PRINT ''"MOVEMENT:";TAB 15;"6 (up)"
  75 PRINT TAB 15;"|"'TAB 15;"|"
  80 PRINT TAB 5;"(west) 5--+--8 (east)"
  85 PRINT TAB 15;"|"'TAB 15;"|"
  90 PRINT TAB 15;"7 (down)"
 100 PRINT AT 19,6; PAPER 6; INK 0;"HIGH SCORE: ";q
 110 PRINT AT 21,4; FLASH 1;"PRESS ANY KEY TO START"
 150 PAUSE 0: CLS
 160 LET u$="UP": LET b=0: LET z$="OFF"
 170 LET d2=(INT (RND*300)-150)*9
 180 LET f1=700: LET t1=5: LET a1=t1*50: LET d1=(INT (RND*100)+300)*100
 190 LET h1=INT (RND*750)+1300: LET c=0: LET z=0
 200 LET a$="████████████████████████████████": GO TO 400
 215 PRINT AT 0,0;"▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄"
 220 FOR a=1 TO 10: PRINT "█                              █": NEXT a
 225 PRINT "▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀"
 227 RETURN
 230 PRINT AT 12,0;a$;"█Speed         ██South        █"
 235 PRINT a$;"█Wheels        ██East         █"
 240 PRINT a$;"█Thrust        ██Altitude     █"
 250 PRINT a$;"█Fuel          ██Brakes       █"
 260 PRINT a$: RETURN
 270 PRINT AT 13,6;"       ";AT 13,22;"       "
 280 PRINT AT 13,14-LEN STR$ a1;a1;AT 13,30-LEN STR$ d1;d1
 285 PRINT AT 15,7;"       ";AT 15,21;"       "
 290 PRINT AT 15,14-LEN u$;u$;AT 15,30-LEN STR$ d2;d2
 295 PRINT AT 17,7;"       ";AT 17,25;"     "
 300 PRINT AT 17,14-LEN STR$ t1;t1;AT 17,30-LEN STR$ h1;h1
 305 PRINT AT 19,5;"        ";AT 19,23;"   "
 310 PRINT AT 19,14-LEN STR$ f1;f1;AT 19,26;z$
 390 RETURN
 400 GO SUB 210: GO SUB 230: GO SUB 270
 500 LET z$="OFF": IF h1<1000 AND INKEY$="w" OR INKEY$="d" THEN LET u$="DOWN": LET z=z+5: LET a1=a1-3
 503 IF b<>0 THEN LET a1=a1+b*10: LET b=0
 505 IF INKEY$="f" AND f1>0 THEN LET t1=t1+1: LET a1=a1+30: LET b=1: IF a1>500 THEN LET a1=a1-30: GO TO 560
 510 IF INKEY$="s" AND f1>0 THEN LET t1=t1-1: LET a1=a1-30: LET b=-1: IF a1<100 THEN LET a1=a1+30: GO TO 560
 515 IF INKEY$="b" THEN LET a1=a1-3: LET z$="ON": GO TO 560
 520 IF INKEY$="6" THEN LET a1=a1-9.8: LET c=c+1: LET d1=d1-(a1*10/6): GO TO 570
 530 IF INKEY$="7" THEN LET a1=a1+9.8: LET c=c-1: LET d1=d1-(a1*10/6): GO TO 570
 540 IF INKEY$="5" THEN LET d1=d1-(a1*.5):: LET d2=d2-(a1*.1): GO TO 570
 550 IF INKEY$="8" THEN LET d1=d
```

```
1-(a1*.5):: LET d2=d2+(a1*.1): G
O TO 570
 560 LET d1=d1-(a1*100/60)
 570 IF t1>10 THEN LET t1=10
 572 IF t1<=0 THEN LET a1=a1-5:
LET t1=0
 575 LET f1=f1-(t1*1.5): LET h1=
h1+(9.8*c*(a1/250)): IF a1>500 T
HEN LET a1=a1-10
 577 IF f1<0 THEN LET f1=0
 578 IF h1<0 THEN LET h1=0
 580 IF c<0 THEN LET c=c+.5
 582 IF c>0 THEN LET c=c-.5
 584 IF a1>t1*50 THEN LET a1=a1-
(a1*.005)
 595 LET d1=INT d1: LET d2=INT d
2: LET f1=INT f1: LET a1=INT a1:
 LET h1=INT h1
 596 GO SUB 270
 597 IF f1<=30 THEN GO SUB 1000
 598 IF NOT (d1<2000 AND ABS d2>
100) THEN PRINT AT 21,10;"
  "
 600 IF d1<2000 AND ABS d2>100 T
HEN GO SUB 1100
 603 IF a1<100 THEN GO SUB 1200
 605 IF h1<150 THEN GO SUB 1300:
 GO TO 619
 610 IF NOT (d1<2000 AND d2>100)
 THEN GO SUB 210
 615 GO SUB 4000
 619 IF d1<0 AND h1>0 AND ABS d2
<100 THEN GO TO 5400
 680 GO TO 500
1000 PRINT AT 21,0; FLASH 1;"LOW
 FUEL"
1030 LET z=z-5
1040 IF f1<=0 THEN LET t1=0
1050 RETURN
1110 IF d1<=0 THEN GO TO 5300
1120 PRINT AT 21,10; FLASH 1;"MO
UNTAINS"
1130 PLOT 10,82: DRAW 40,85: DRA
W 25,-85: PLOT 71,95: DRAW 20,25
: DRAW 40,-38: PLOT 125,90: DRAW
 30,35: DRAW 18,-43: PLOT 165,10
5: DRAW 45,40: DRAW 40,-40
1140 LET b=1
1150 PAUSE 30: RETURN
1210 PRINT AT 21,25; FLASH 1;"ST
ALL"
1240 LET h1=h1-35:: LET t1=0: LE
T a1=a1+28: IF h1<=0 THEN GO TO
5200
1250 FOR a=0 TO 100: NEXT a: PRI
NT AT 21,25;"     ": RETURN
1310 IF d1>=5000 THEN GO TO 1400
1320 IF d1<5000 AND ABS d2>100 T
HEN GO TO 1500
1330 IF d1<5000 AND d1>=3000 AND
 ABS d2<=100 THEN GO TO 1600
1340 IF d1<3000 AND d1>=1800 AND
 ABS d2<=100 THEN GO TO 1700
1350 IF d1<1800 AND d1>=1200 AND
 ABS d2<=100 THEN GO TO 1800
1360 IF d1<1200 AND d1>=600 AND
ABS d2<=100 THEN GO TO 1900
1370 IF d1<600 AND d1>=0 AND ABS
 d2<=100 THEN GO TO 2000
1400 PRINT AT 1,0;b$
1410 IF h1<=0 THEN GO TO 5000
1420 GO TO 3000
1500 IF d1>500 THEN GO TO 1510
1505 GO SUB 210
1520 GO TO 3000
1600 PRINT AT 1,0;b$: LET z=z+5
1620 PRINT AT 2,d2/12.5+14;"/ \"
: IF h1<=0 THEN GO TO 5000
1630 GO TO 3000
1700 PRINT AT 1,0;b$: LET z=z+5
1720 PRINT AT 2,d2/12.5+14;"/ \"
;AT 3,d2/12.5+13;"/   \"
1730 GO TO 3000
1800 PRINT AT 1,0;b$: LET z=z+5
1820 PRINT AT 2,d2/12.5+12;"/
\";AT 3,d2/12.5+11;"/       \";AT
 4,d2/12.5+10;"/         \"
1830 IF h1<=0 THEN GO TO 5000
1840 GO TO 3000
1900 PRINT AT 1,0;b$: LET z=z+10
1920 PRINT AT 2,d2/12.5+11;"/ |
 \";AT 3,d2/12.5+10;"/   |   \"
;AT 4,d2/12.5+9;"/     |     \";AT
 5,d2/12.5+8;"/       |       \"
1940 IF h1<=0 THEN GO TO 5000
1950 GO TO 3000
2000 PRINT AT 1,0;b$: LET z=z+15
2030 PRINT AT 2,d2/12.5+11;"/
|   \";AT 3,d2/12.5+10;"/     |
  \";AT 4,d2/12.5+9;"/       |
 \";AT 5,d2/12.5+8;"/         |
  "
2040 PRINT AT 6,d2/12.5+7;"/
   |        \";AT 7,d2/12.5+6;"/
          |          \"
2050 IF h1<=0 THEN GO TO 5060
3000 RETURN
4000 RETURN
5010 PAUSE 100: FOR a=12 TO 21:
PRINT AT a,0;"
                 ": NEXT a: PRINT A
T 12,0;
5020 PRINT "YOU CRASHED AT ";A1;
" KMH,": LET Z=Z-70
5030 PRINT D1;" METRES FROM THE
AIRPORT,"
5035 PRINT "AND ";ABS D2;" METRE
S OFF CENTRE."
5040 PRINT "THE PLANE IS A WRITE
OFF."
5050 PRINT "THERE WERE NO SURVIV
ORS.": LET Z=Z-100: GO TO 6000
5060 PAUSE 200
5070 IF ABS D2<20 THEN PRINT "YO
U LANDED ON THE RUNWAY ";D2
5075 IF ABS D2<20 THEN PRINT "ME
TRES OFF COURSE AND ";D1
5080 IF ABS D2<20 THEN PRINT "ME
TRES FROM THE ";: LET Z=Z+30
5085 IF ABS D2>20 THEN PRINT "YO
U JUST MISSED THE RUNWAY:"
5090 IF ABS D2>20 THEN PRINT "YO
U WERE ";D2;"METRES"
5095 IF ABS D2>20 THEN PRINT "OF
F COURSE AND ";D1;" METRES"
5100 IF ABS D2>20 THEN PRINT "FR
OM THE ";
5105 PRINT "AIRPORT."'"YOUR SPEE
D WAS ";A1;" KMH."
5110 IF U$<>"DOWN" THEN PRINT "Y
OUR UNDERCARRIAGE WAS NOT DOWN:T
HE PLANE WAS A WRITEOFF.": LET Z
=Z-50
5115 IF A1<250 THEN GO TO 6000
5120 LET D=INT ((A1-250)/20): FO
R E=1 TO D: LET F=INT (RND*10)
5125 IF F=7 THEN PRINT "YOU LAND
ED TOO FAST,THE PLANE   JUST BRO
KE UP COMPLETELY.": LET Z=Z-80:
GO TO 6000
5150 NEXT E: GO TO 6000
5200 PRINT AT 12,0;"YOUR PLANE S
TALLED AND CRASHED  ON IT'S NOSE
,NOBODY SURVIVED.   YOU WERE ";A
BS D2;" METRES",
5210 PRINT "OFF COURSE AND ";D1;
" METRES","FROM THE AIRPORT.",
5220 LET Z=Z-150: GO TO 6000
5300 PRINT AT 12,0;"YOU SMASHED
INTO A MOUNTAIN AT"'A1;" KMH.",,
: LET Z=Z-200
5320 PRINT "YOU WERE ";ABS D2;"
METRES OFF  COURSE.",,: GO TO 60
00
5400 PRINT AT 12,0;"YOU FLEW RIG
HT OVER THE AIRPORT.": LET Z=Z-5
0
6060 LET Z=Z+INT ((200/(ABS D1+1
))+(150/(ABS D2+2))+(F1/15)+(30-
(H1/150)))
```

```
6065 IF Z<0 THEN LET Z=0
6070 PRINT "YOU SCORED ";Z;" POI
NTS.",: IF Z>Q THEN LET Q=Z
6075 PRINT ,,"HIGHEST SCORE: ";Q
,
6080 PRINT ,,"PRESS SPACE TO CON
TINUE.",
6090 IF INKEY$<>" " THEN GO TO 6
090
6100 GO TO 10
8000 FOR a=USR "a" TO USR "a"+23
: READ w: POKE a,w: NEXT a
8010 DATA 1,2,4,8,16,32,64,128
8020 DATA 128,64,32,16,8,4,2,1
8030 DATA 0,32,32,112,112,248,24
8,32
8040 RETURN
```

Some sample screen dumps from the game, Pilot – as you can see, it's not as easy as you might think!



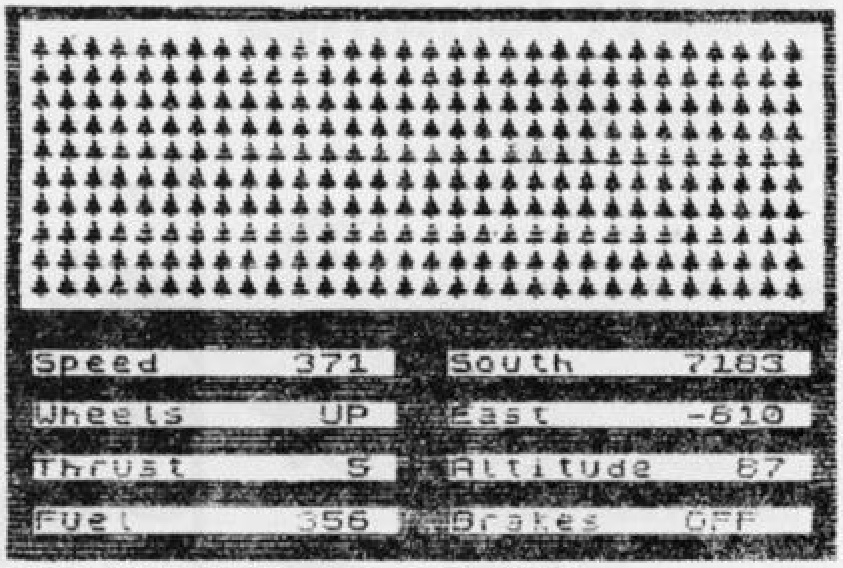
MOUNTAINS

YOU SMASHED INTO A MOUNTAIN AT
350 KMH.
YOU WERE 616 METRES OFF COURSE.

YOU SCORED 0 POINTS.

HIGHEST SCORE: 0

PRESS SPACE TO CONTINUE.
MOUNTAINS



YOU CRASHED AT 367 KMH,
5949 METRES FROM THE AIRPORT,
AND 810 METRES OFF CENTRE.
THE PLANE IS A WRITEOFF.
THERE WERE NO SURVIVORS.
YOU SCORED 0 POINTS.

HIGHEST SCORE: 0

PRESS SPACE TO CONTINUE.



The ZX81 at play

A whole host of games programs for your 1K ZX81.

Ladder landings

In this game, Zakariya Ahad has those lovable aliens climbing down to Earth on ladders! Having journeyed across the stars, the alien force find that the only way they can get down to the Earth's surface is to construct four ladders from their mother ship.

Using the keys, 'Z' and 'M', you have to guide a sleek spaceship and stop the aliens from bringing the ladders down to the ground. Once established, the ladders can be used to bring more aliens down from their mother ship. If too many aliens land, you lose the game. See how long you can defend the Earth.

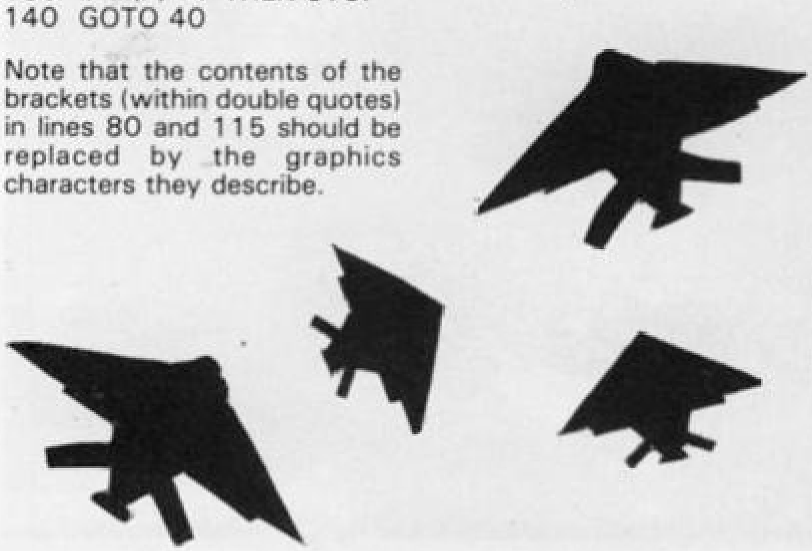
The listing is as follows.





```
10  LET H=11
30  DIM A(4)
40  LET P=INT(4 * RND)+1
60  LET J=5 * P
70  FOR I=10 TO 21-A(P)
80  PRINT AT I,J;" (Shift Graphics F and G)"
81  PRINT AT I-1,J;" "
90  LET M$=INKEY$
100 IF M$="Z" THEN LET H=H-1
110 IF M$="M" THEN LET H=H+1
115 PRINT AT 15,H;" (space,Shift Graphics W, inverse
    D,Shift Graphic Q, space)"
120 IF I=15 AND ABS(H+2-J) <=1 THEN GOTO 40
125 NEXT I
130 LET A(P)=A(P)+1
135 IF A(P)=7 THEN STOP
140 GOTO 40
```

Note that the contents of the brackets (within double quotes) in lines 80 and 115 should be replaced by the graphics characters they describe.

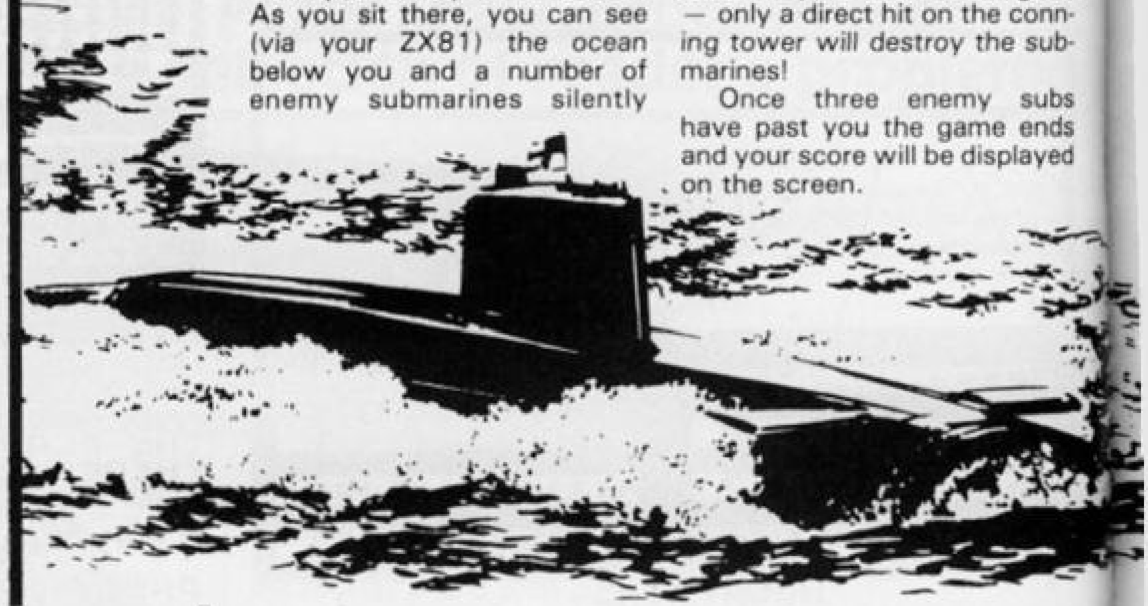


Subcharge

Twelve year old Jonathon Almond shares a novel naval program with us. You control a destroyer which remains stationary at the top of the screen. As you sit there, you can see (via your ZX81) the ocean below you and a number of enemy submarines silently making their way past your defences.

Using the '0' key you can drop depth charges onto the subs but your aim must be good — only a direct hit on the conning tower will destroy the submarines!

Once three enemy subs have past you the game ends and your score will be displayed on the screen.



```
 10 REM SUBCHARGE
 15 RAND
 20 LET L=3
 30 LET SC=0
 40 LET DF=0
 45 LET S=INT (RND*13)+7
 50 LET D=2
 60 FOR A=27 TO 0 STEP -1
 65 PRINT AT 1,7;"-▀▙/▀ /";AT 2,
6;"▄▄▄▄▄▄▄▄▄▄"
 70 PRINT AT S,A;"▄▄▟▄▄▄▄ "
 75 IF INKEY$="0" THEN LET DF=1
 80 IF DF=1 THEN GOTO 100
 85 NEXT A
 90 GOTO 200
100 PRINT AT D,10;" "
110 LET D=D+1
120 PRINT AT D,10;"O"
125 IF D=S AND A+2=10 THEN GOTO
400
130 IF D>S THEN GOTO 300
140 GOTO 65
200 LET L=L-1
205 LET DF=0
210 CLS
220 IF L>0 THEN GOTO 45
230 PRINT AT 11,11;"GAME OVER"
240 PRINT AT 12,11;"SCORE: ";SC
250 STOP
300 LET DF=0
305 PRINT AT D,10;" "
310 LET D=0
320 GOTO 85
400 PRINT AT S,A;"EO=OE"
410 LET SC=SC+1
415 LET DF=0
420 CLS
430 GOTO 45
```



The golfing guillotine

Not such a silly title when you consider the double bill written for us by Grahame Chidwick of Grimsby. In his first program called Guillotine, a 'Hangman' type game, you have to guess a word in eight goes or else some heads start rolling!

At the prompt, the first player types in a word, say the name of an object in the room or the title of a movie, etc, while the second player looks away. Player two now has to input individual letters or make a guess at the word. However, you only have eight attempts at the word — so don't lose your head.

In Grahame's second program called Computer Golf, a simulation of 18 holes of golf, you have to guess the distance from you to the hole.

When the game starts you are sited at the left-hand side of the green. To hit your ball towards the hole, you simply type in a positive value for the distance you wish your ball to travel. Should you overshoot the hole, you must imput a negative value for distance.

The par for each hole is displayed and your score is updated at the end of each hole. A negative score means you are under par and a positive score means that you probably aren't in the Jack Nicklaus class just yet.

Guillotine

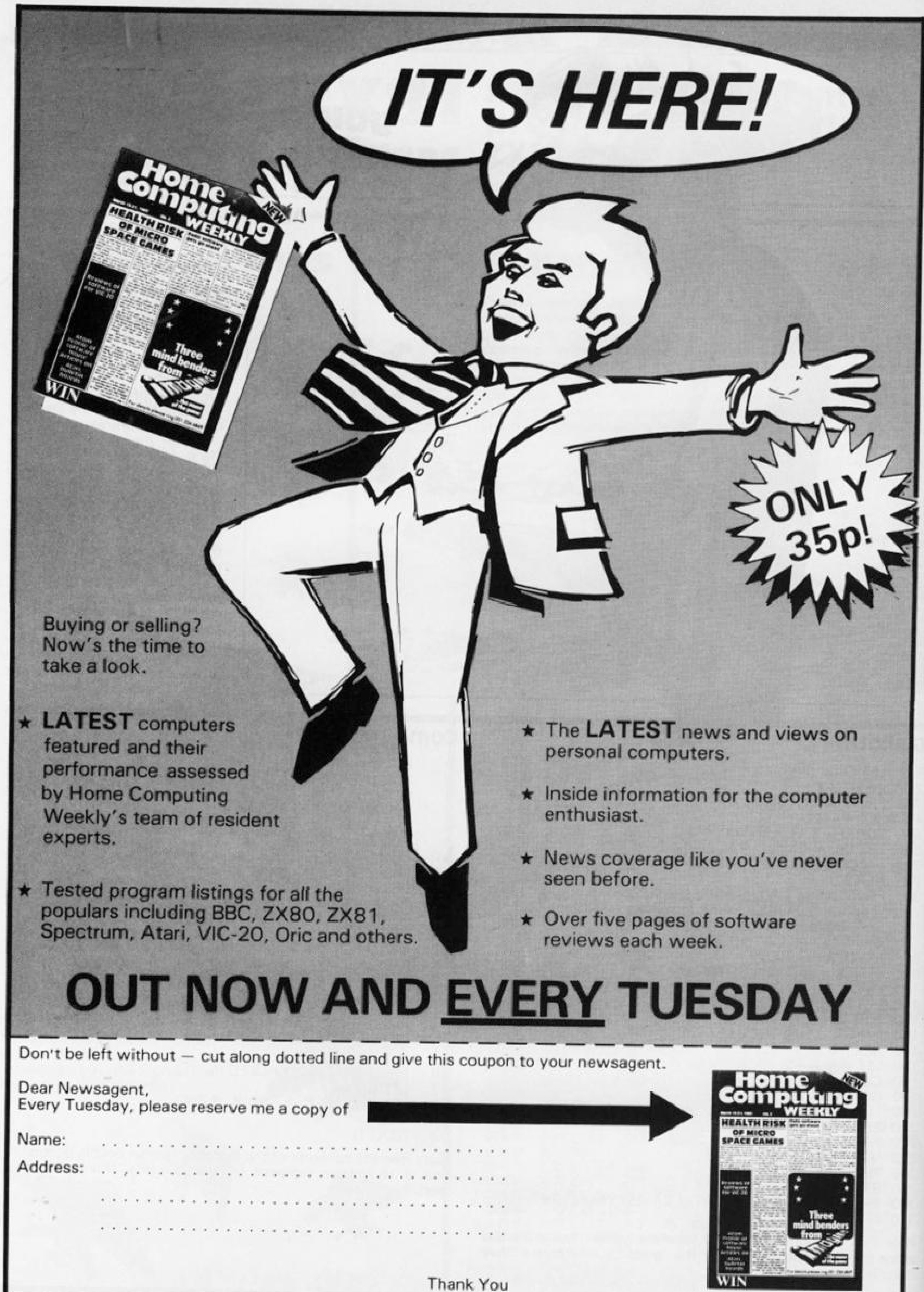
```
10  LET C$ = " "
20  INPUT A$
30  PRINT AT 1,15;LEN A$
40  LET A=0
50  FOR D= 1 TO LEN A$
60  LET C$ = C$ +" *"
65  IF CODE A$(D) >CODE "Z" THEN LET C$(D) = A$(D)
70  NEXT D
80  PRINT AT A,1;"(5 Graphic spaces)";AT A+1;1; "(Shift
    Graphics E, Shift Graphics 1, space, Shift Graphics 2,
    Shift Graphics R)"; AT 16,3;"O"; AT 17,1; "(5 Shift
    Graphics As)"; AT 19,3; "U"
81  PRINT AT 0,15;C$
90  IF A >=16 THEN GOTO 200
100 INPUT B$
110 IF A$= B$ THEN GOTO 300
120 FOR D= 1 TO LEN A$
130 IF B$= A$(D) THEN LET C$(D)=B$
140 NEXT D
150 LET A= A+2
160 CLS
170 GOTO 80
200 FOR D=16 TO 19
210 PRINT AT D—1,3; " "
220 PRINT AT D,3 ; "O"
230 NEXT D
240 PRINT A$
250 STOP
300 PRINT "SCORE=";A, "YOU SAVED YOUR LIFE"
310 PRINT A$
```

Note that the contents of the brackets (within double quotes) in line 80 should be replaced by the graphics characters they describe.

Computer golf

```
2   LET S=0
3   FOR N=1 to 18
4   LET O=1
5   LET A=INT(RND * 23)+4
6   LET T=A
7   FOR Z=1 TO 100
9   PRINT AT 20,0;"(32 Graphic spaces)"
10  PRINT AT 0,0;"HOLE: ";N
11  LET P=INT(A/10+.5)+1
12  PRINT AT 5,0;"PAR: ";P
13  PRINT AT 8,0;"SCORE= ";-S
16  INPUT B
17  LET O=O+B+(INT(RND * 5)-2) AND B RND * 5 OR
    RND (A-O)/O
18  PRINT AT 19,0;"O"
20  IF O=A THEN GOTO 24
21  LET X=RND * *RND * *RND
22  PRINT AT 19,0; " "
23  NEXT Z
24  LET K=P-Z
26  PRINT AT 18,0; "HOLED IN: ";Z
27  LET S=S+K
28  LET X=RND * *RND * *RND
32  CLS
33  NEXT N
```

Note that the contents of the brackets (within double quotes) in line 9 should be replaced by the graphics characters they describe.


IT'S HERE!
Home Computing WEEKLY
NEW
HEALTH RISK OF MICRO SPACE GAMES
Three mind benders from
WIN
ONLY 35p!
Buying or selling? Now's the time to take a look.
★ LATEST computers featured and their performance assessed by Home Computing Weekly's team of resident experts.
★ Tested program listings for all the populars including BBC, ZX80, ZX81, Spectrum, Atari, VIC-20, Oric and others.
★ The LATEST news and views on personal computers.
★ Inside information for the computer enthusiast.
★ News coverage like you've never seen before.
★ Over five pages of software reviews each week.
OUT NOW AND EVERY TUESDAY
Don't be left without — cut along dotted line and give this coupon to your newsagent.
Dear Newsagent,
Every Tuesday, please reserve me a copy of
Name:
Address:
Thank You

Home Computer Show

Birmingham

THE METROPOLE HOTEL AT THE NATIONAL EXHIBITION CENTRE

VISIT THE COMPLETE SHOW FOR THE HOME USER AND SEE:

A COMPLETE cross section of all hardware and software available to the home user.

A FULL RANGE of home computers priced from £50 upwards.

A COMPUTER ADVICE CENTRE run by independent experts for the answers to all our questions.

WIN WIN WIN TWO COMPUTERS — one for you, one for a school of your choice — to be won at each show: FREE entry form with advance tickets

JUNE SATURDAY 4th (10am-6pm)
SUNDAY 5th (10am-4pm)

ADULTS: £2.00 UNDER 8s & OAPs: FREE
25% DISCOUNT for parties of 20 or more.

SPONSORED JOINTLY BY:

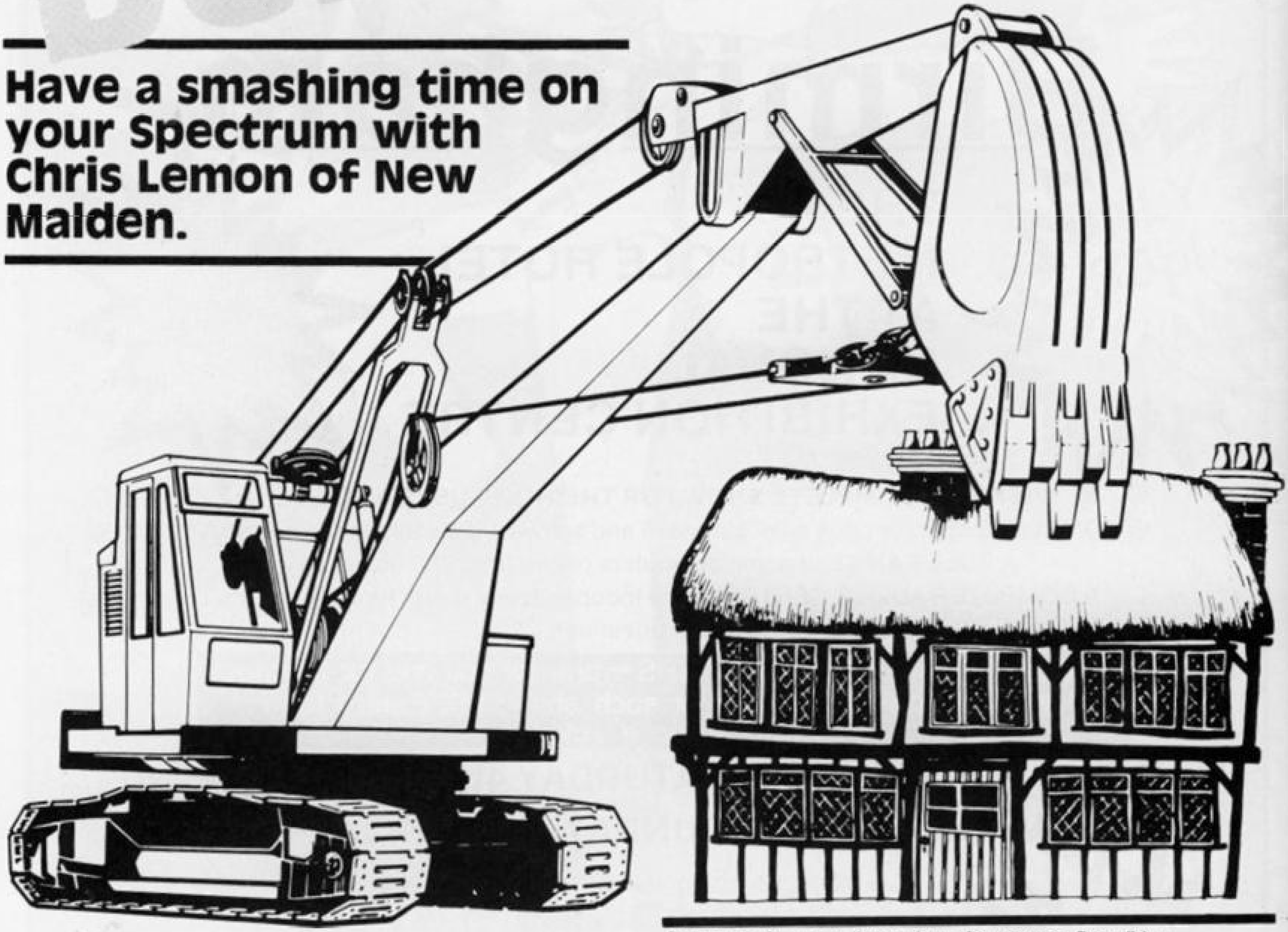
A&B Computing
Computing Today Personal Software
Personal Computing Today
Home Computing Weekly
ZX Computing

BIRMINGHAM
Home Computer Show
Send £1.00 Save £1.00 One person per coupon only
Please enclose SAE
Mr. Mrs. Miss
Address

ASP Exhibitions
145 Charing Cross Road
London WC2H 0EE
01-437 1002

Demolition

Have a smashing time on your Spectrum with Chris Lemon of New Maiden.



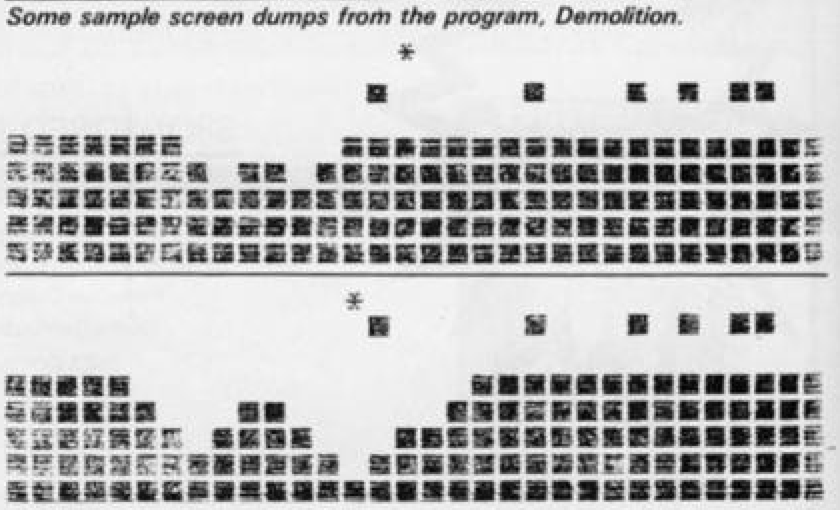
In this game for your 16K Spectrum, the player is presented with a series of brick walls which slowly move up the screen. The walls must be demolished by dropping an explosive asterisk down onto sections of the walls. The game is over when the wall reaches the top of the screen and the asterisk bumps into it.

The program has two features:
- The score is printed in the input area.
-The walls are advanced by scrolling with SCR CT supressed.

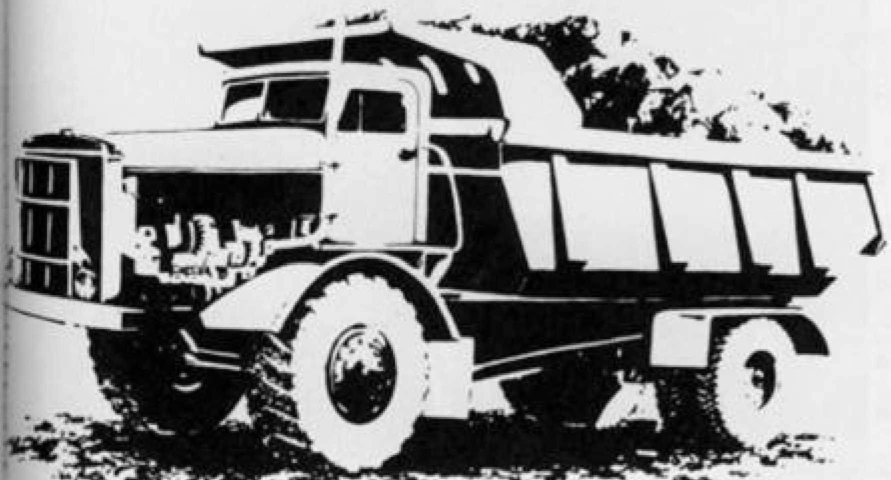
The directions for play are very simple - once typed in and RUN, you simply press Enter to start the game and press the '6' key to drop the explosive asterisk as it moves across the top of the screen. The asterisk can move from right to left or *vice versa;* there is no warning given as to which side the asterisk will come.

Demolition is a simple, though very addictive game.

Some sample screen dumps from the program, Demolition.



```
  40 RANDOMIZE
  50 BORDER 6
  60 PRINT AT 10,6;"D E M O L I T I O N"
  70 PRINT AT 15,4;"Use 6 to break the walls."
  80 PRINT AT 16,4;"Press enter to start."
  90 INPUT LINE l$
  95 CLS
 100 LET c=0: LET k=8: LET a=0: LET s=0: LET p=1
 190 GO TO 8000
 200 REM
 202 GO SUB 4000
 210 FOR i=0 TO 31
 211 IF SCREEN$ (1,i)="" THEN GO TO 3000
 213 NEXT i
 220 LET c=c+1: IF c=k THEN GO TO 1000
 240 LET vv=INT (RND*2)
 250 IF vv=0 THEN GO TO 300
 260 GO TO 350
 300 REM go right
 310 PRINT AT 1,0;"*";
 312 IF INKEY$="6" THEN GO TO 500
```



```
 320 FOR i=1 TO 31
 322 BEEP .01,16: PAUSE p
 330 PRINT CHR$ 8;" ";" ";CHR$ 8;"*";
 332 IF INKEY$="6" THEN GO TO 500
 340 NEXT i
 342 PRINT CHR$ 8;" ";
 348 GO TO 200
 350 REM go left
 360 PRINT AT 1,31;"*";
 362 IF INKEY$="6" THEN GO TO 500
 370 FOR i=0 TO 30
 372 BEEP .01,16: PAUSE p
 380 PRINT CHR$ 8;" ";CHR$ 8;CHR$ 8;"*";
 382 IF INKEY$="6" THEN GO TO 500
 390 NEXT i
 392 PRINT CHR$ 8;" ";
 398 GO TO 200
 500 REM drop
 510 LET x=33-PEEK 23688-1
 520 FOR i=2 TO 20
 530 PRINT AT i-1,x;" ";
 540 IF SCREEN$ (i,x)="" THEN GO TO 700
 542 PRINT AT i,x;"*";: BEEP .01,8
 550 NEXT i
 552 PRINT AT 20,x;" ";
 560 GO TO 200
 700 REM percu
 710 PRINT AT i,x;" ": LET s=s+10: BEEP .02,16
 712 LET i=i+1: IF i>21 THEN GO TO 720
 714 IF SCREEN$ (i,x)="" THEN LET s=s+10
 716 PRINT AT i,x;" "
 720 FOR j=1 TO 5
 722 LET xm=x-j: IF xm<0 THEN LET xm=0
 724 LET xp=x+j: IF xp>31 THEN LET xp=31
 726 LET sm=(SCREEN$ (i-j,xm)="")
 727 LET tm=(SCREEN$ (i-j-1,xm)="")
 728 LET sp=(SCREEN$ (i-j,xp)="")
 729 LET tp=(SCREEN$ (i-j-1,xp)="")
 730 LET w=sm+sp+tm+tp
 740 IF w=0 THEN GO TO 200
 760 PRINT AT i-j,xm;" ";: PRINT AT i-j,xp;" ";
 770 PRINT AT i-j-1,xm;" ";: PRINT AT i-j-1,xp;" ";
 772 LET s=s+w*10: BEEP .05,4
 780 NEXT j
 810 GO TO 200
1000 LET c=0: LET a=a+1: IF a>5 THEN GO TO 1020
1010 POKE 23692,255: PRINT AT 21,31;b$: PRINT AT 20,0;b$: GO TO 1100
1020 LET q=INT (RND*7)
1022 POKE 23692,255: PRINT AT 21,31;b$: PRINT INK q;AT 20,0;a$: IF a=10 THEN LET a=0
1030 GO TO 1100
1100 LET k=k-1: IF k=2 THEN LET k=3
1110 GO TO 200
2100 LET a$="GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG"
2110 LET b$="                                "
2112 LET z$="         .....AAAARGHHHH....."
2120 FOR i=0 TO 15
2130 PRINT b$
2140 NEXT i
2150 FOR i=16 TO 20
2160 PRINT INK (i-16);a$
2170 NEXT i
2180 PRINT b$
2190 GO TO 200
3000 REM stop
3010 PRINT AT 21,0;z$
3020 PAUSE 0
3024 PRINT AT 0,0;s
3030 STOP
4000 REM score
4020 LET s$=STR$ s: LET l=LEN s$
4030 FOR j=1 TO l
4040 LET f=CODE s$(j)
4050 FOR i=0 TO 7: LET e=PEEK (u+f*8+i): POKE (v+j-1+256*i),e: NEXT i
4060 NEXT j
4100 RETURN
8000 LET g0=BIN 00000000
8010 LET g1=BIN 01111110
8020 POKE USR "g"+0,g0
8030 FOR i=1 TO 6
8040 POKE USR "g"+i,g1
8050 NEXT i
8060 POKE USR "g"+7,g0
8080 LET u=PEEK (23607)*256+PEEK (23606)
8090 LET v=16*256+32*6+16384
8100 GO TO 2100
```

Pinball

Anyone can become a pinball wizard with this program from Mark Pattinson from Tyne and Wear.

Once you've typed this program into your 16K ZX81, you can start the game by pressing any key.

The ball will move around the pintable, deflecting off 'bumpers', the sides of the table and the bat (which you control). The ball may fall into one of the holes ('O') and then bounce out, or be deviated in a random direction when it touches the wall of 'X's.

You score points in the following way:

Deflecting the ball off the bat – 50 points.
Landing in one of the holes – 100 points.
Deflecting the ball from the wall of 'X's – 10 points.

Should you manage to score 2,000 points, you'll find that the bat you use to deflect the ball back on the table reduces in size from three inverse spaces to just two! To move the bat, you use the 'M' key to move right and the 'Z' key to move left.

Your score is incremented as you deflect off objects, and the ball's speed will increase in proportion to your score.

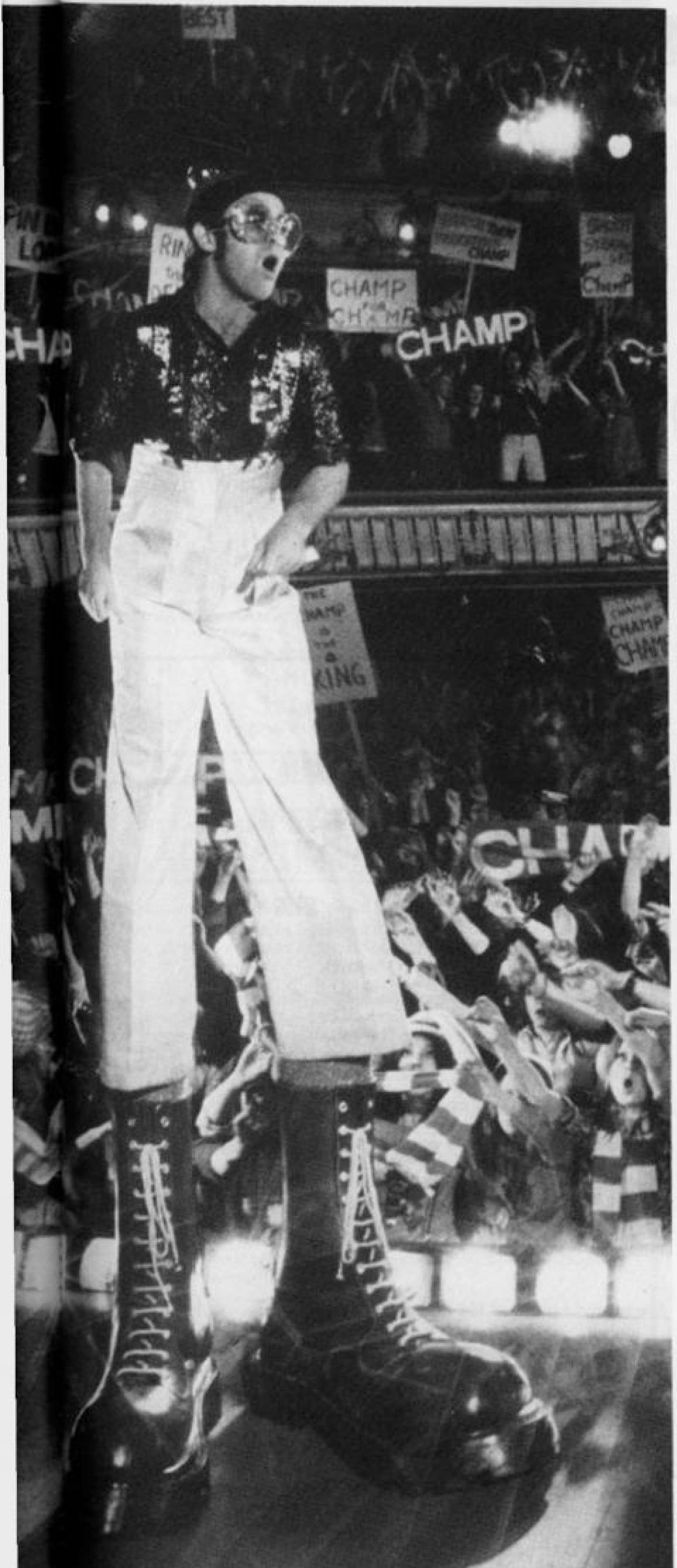
What's your line?

10-195 Variables and board are PRINTed.
215-301 The ball is moved up along the chute and drops through one of the four gates, which close immediately afterwards.
320-360 The main loop of the program.
380 Alters the horizontal velocity.
430 Alters the vertical velocity.
600-630 End of the game. PRINTs the high score, etc, and re-starts the game.
850-855 Ball falls into a hole and bounces out.
940-955 Ball bounces off the line of 'X's.
1610-1615 Ball bounces off the bat.
2070 Ball bounces off a corner of the table.
5000 Alteration of the bat size once you reach 2,000 points.

```
   1 LET N$=""
   2 LET S=0
   3 REM ********█████████********
   4 REM
   5 REM █-DOWN**************█-UP
  10 RAND
  15 LET A$="..████.."
  20 LET SC=0
  30 LET A=10
  40 LET Y=20
  50 LET X=30
  60 CLS
  70 PRINT "█ PINBALL     █████=
█"
  80 PRINT "█████████████████████████
███████████"
  90 PRINT "█";TAB 31;"█"
 100 FOR I=1 TO 18
 110 PRINT "█";TAB 29;"█";TAB 31
;"█"
 120 NEXT I
 130 PRINT AT 3,1;"████████ ████ █
█ ███ ██████████"
 139 PRINT AT 5,6;"O       O     O
 O"
 140 PRINT AT 6,2;"O     O      O
O     O    O"
 141 PRINT AT 7,2;"O";TAB 25;"O"
 150 PRINT AT 8,4;"O       O
O     O"
 151 PRINT AT 9,4;"O       O
O     O"
 155 PRINT TAB 12;"█████"
 160 PRINT AT 13,5;"████";TAB 20;
"████"
 165 PRINT AT 11,2;"XXXXXXXXXXXX
XXXXXXXXXXXXXX"
 166 PRINT AT 3,0;"█";AT 3,29;"█
"
 170 PRINT AT 21,0;"..............
......................"
 180 PRINT AT 20,0;"████";TAB 27;
"████"
 195 PRINT AT 21,A;"..████..";AT
Y,X;"0"
 210 IF INKEY$="" THEN GOTO 210
 211 IF INKEY$="" THEN GOTO 211
 215 FOR Y=Y TO 3 STEP -1
 220 PRINT AT Y,X;"0"
 225 PRINT AT Y,X;" "
 230 NEXT Y
 240 LET QQ=INT (RND*4)*5+6
 250 FOR X=X TO QQ+1 STEP -1
 260 PRINT AT Y,X;"0"
 265 PRINT AT Y,X;" "
 270 NEXT X
 280 LET DY=1
 290 LET DX=0
 300 PRINT AT 3,1;"████████████████
████████████"
 301 LET Y=Y+1
 315 PRINT AT 0,18;SC
 320 LET Y=Y+DY
 321 LET X=X+DX
 325 GOTO PEEK (PEEK 16396+256*P
EEK 16397+1+Y*33+X)*10+330
 330 PRINT AT Y,X;"0"
 340 LET A=A+2*((INKEY$="M" OR A
```

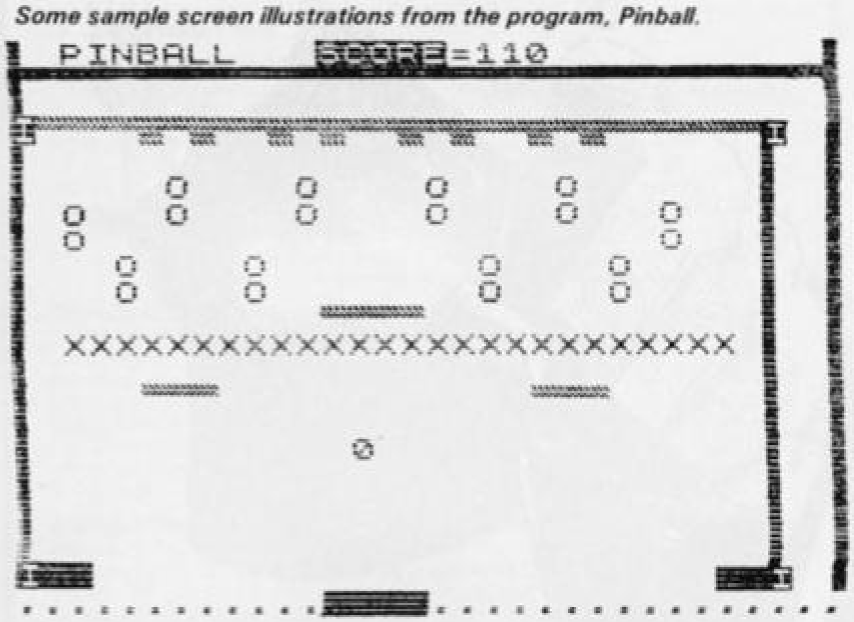


The Robert Stigwood Group Ltd.




```
=0)-(INKEY$="Z" OR A=24))
 350 PRINT AT 21,A;A$;AT Y,X;" "
 360 GOTO 320
 380 LET DX=-DX
 381 GOTO 320
 430 LET DY=-DY
 431 GOTO 320
 600 IF SC>S THEN INPUT N$
 605 IF SC>S THEN LET S=SC
 610 PRINT AT 15,1;"  HI-SCORE B
Y ";N$
 611 PRINT TAB 10;S
 615 PAUSE 4E4
 620 IF INKEY$="" THEN GOTO 620
 630 GOTO 10
 850 PRINT AT Y,X;"0"
 851 LET DX=-DX+(DX=0)*(INT (RND
*2)*2-1)
 852 LET SC=SC+100
 854 PRINT AT Y,X;"O"
 855 GOTO 315
 940 PRINT AT Y,X;"0"
 945 LET SC=SC+10
 950 LET DX=INT (RND*2)*2-1
 952 PRINT AT Y,X;"X"
 955 GOTO 315
1610 LET DY=-1
1611 LET SC=SC+50
1612 IF SC>2000 AND LEN A$>6 THE
N GOTO 5000
1615 GOTO 315
2070 LET DY=-DY
2071 LET DX=-DX
2072 GOTO 320
5000 PRINT AT 21,A;"........"
5010 LET A$="....."
5020 GOTO 315
```

Some sample screen illustrations from the program, Pinball.





No man's land

Mark Emery of Surrey has written a fine program for you and your ZX81.

```
10 REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
   XXXXXXXXXXXXXXXXXXXXXXX
20 LET A=16514
30 INPUT B
40 POKE A, B
50 PRINT B;",";
60 LET A=A+1
70 GOTO 30
```

The machine code loader program.

```
17/0/0/98/107/175/125/50/164/64/175/124/50/165/
64/1/214/2/237/66/200/98/107/1/33/0/9/84/93/237/
75/12/64/33/0/0/9/175/60/35/35/254/32/40/7/70/54/
0/43/112/24/242/24/205/
```

The machine code part of the listing.

It should be noted that a '/' denotes Newline.



```
15  LET F=O
20  LET E=14
30  LET I=O
40  PRINT AT INT (RND * 22),30;"■■";AT INT
    (RND * 22),31;"O"
50  PRINT AT E,14;" = + )";
60  IF PEEK (PEEK 16398+256 * PEEK 16399)=128
    THEN GOTO 130
70  IF PEEK (PEEK 16398+256 * PEEK 16399)=52 THEN
    LET I=I+1
80  PRINT AT INT (RND * 22),30;"■■"
90  LET A=USR 16514
100 PRINT AT E,13;"          "
110 LET E=E+(INKEY$="O" AND
    E < 21)-(INKEY$="1" AND E >O)
120 GOTO 40
130 LET F=F+1
140 PRINT AT E,14;") * ("
145 PAUSE 75
147 CLS
150 IF F >3 THEN PRINT AT 11,10;"GAME OVER";AT
    13,10;"SCORE :";END
160 PRINT AT 11,10;"SHIP ";F;AT 13,10;" SCORE SO
    FAR :";I
165 FOR B=1 to 100
167 NEXT B
170 CLS
180 GOTO 40
```

The second section of the listing.

You are in a small patrol ship patrolling the area of no man's land of a battle zone in the future. As you move across the terrain, you come across many mines (shown as black squares) placed in your path. You must move to avoid these (by pressing the '1' and 'O' keys) otherwise they will explode destroying your ship. You will also see on the screen the positions of snipers (shown as Os). Although they will not harm you, your prime objective is to ram them thereby killing them. Every sniper you hit and destroy will be added to your score. You have four ships and after all of them have been destroyed, the game will end and your score will be displayed.

The program is entered in two parts. The first part is used to load the machine code routine. When RUN, the computer will ask you for a number. You must then enter the machine code, each number being followed by Newline. The second section of the listing is the main part of the program and can be entered straight on top of the machine code program. Line 10 of the machine code program will be left and will become part of the second program.

Learner driver

You'll have to watch out for more than the cat's eyes in this game from Mr Davis of Checkley.



20th Century-Fox Film Corp.

All you have to do in this program is to drive your car from the top left of the screen to the bottom right without hitting anything. Sound's easy enough...

When you have got the program RUNning, you find yourself looking at a page of information explaining the rules of the game. Movement of your vehicle, which is placed at the top left-hand side of the screen, is controlled by the '1' key to move up, the 'z' key to move down, the 'm' key to move left and the 'z' key to move right. Then, you have a minute period to manoeuver your car to the bottom left-hand corner gaining as many points as you can.

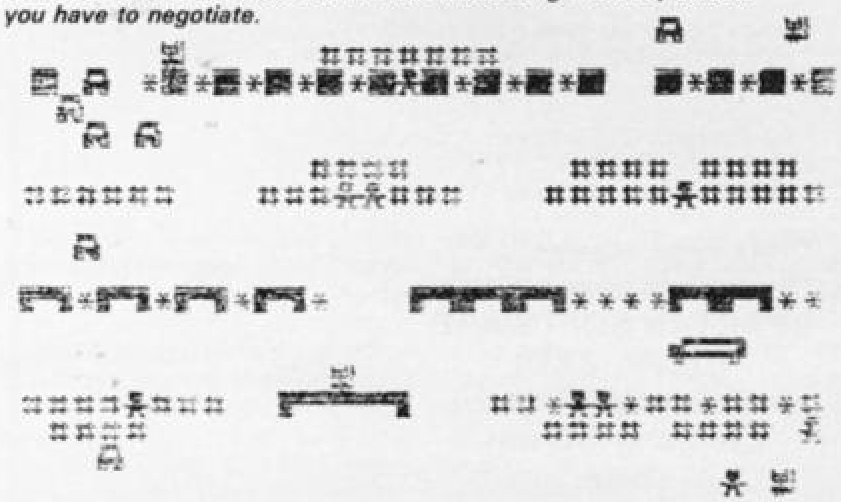
Mind that bollard!

To gain points en route, you can demolish walls and fences, although you mustn't crash into a flashing bollard. Should you run over a pedestrian, or collide with one of the moving cars or trucks, you will also be told that you have crashed, given your score and asked if you would like to have another game.

You are permitted to run over cats, and as a special bonus you are awarded 100 points if you manage to run over the cat in the bottom right of the screen.

So, if you fancy an exciting drive, or you simply don't like cats, then this is the program for you.

A screen illustration from Learner driver showing the busy streets you have to negotiate.



```
   1 INK 0: PAPER 6
   2 LET hsc=0
   5 DATA 138,250,170,250,34,250
,254,80
  10 DATA 60,60,60,24,255,60,36,
102
  15 DATA 15,15,248,248,255,255,
40,56
  20 DATA 126,126,66,195,255,255
,165,231
  25 DATA 255,255,0,0,255,255,0,
0
  30 DATA 255,255,7,7,255,255,20
,28
  40 LET u=PEEK 23675+256*PEEK 2
3676
  50 FOR i=0 TO 7
  52 READ j
  54 POKE u+16+i,j
  56 NEXT i
  60 FOR i=0 TO 7
  62 READ j
  64 POKE u+40+i,j
  66 NEXT i
  70 FOR i=0 TO 7
  72 READ j
  74 POKE u+120+i,j
  76 NEXT i
  80 FOR i=0 TO 7
  82 READ j
  84 POKE u+128+i,j
  86 NEXT i
  90 FOR i=0 TO 7
  92 READ j
  94 POKE u+136+i,j
  96 NEXT i
 100 FOR i=0 TO 7
 102 READ j
 104 POKE u+144+i,j
 106 NEXT i
 120 PRINT AT 1,8;"LEARNER DRIVE
R";AT 3,2;"Drive from top left t
o bottom  right,demolishing all
walls and fences. Avoid collidin
g with    moving vehicles, pedes
trians,   and flashing bollards,
 although running over cats is p
ermitted. The final cat scores a
 bonus    100 points."
 121 OUT 254,INT (RND*32): PAUSE
 300
 122 PRINT AT 12,6;"Time Limit:
2 min."
 124 PRINT AT 14,2;"Controls  UP
  '1'    LEFT 'M'"
 126 PRINT AT 16,11;"DOWN 'Z'  R
IGHT 'P'"
 128 PRINT AT 19,7;"Press 'P' to
 play"
 130 IF INKEY$="p" THEN GO TO 15
0
 135 GO TO 120
 150 CLS
 160 DEF FN t()=INT ((65536*PEEK
 23674+256*PEEK 23673+PEEK 23672
)/50)
 165 LET t1=FN t()
 170 LET s=0
 177 LET x=0
 178 LET y=0
 189 PRINT INK 2;AT 17,5;"F";AT
17,22;"FF";AT 20,28;"F";AT 3,15;
"F";AT 6,13;"FF";AT 6,26;"F";AT
16,31;"F": REM GRAPHICS f
 190 PRINT AT 3,0;"█";AT 3,5;"*
█*█*█*█*█";AT 3,16;"█*█*█*█";AT
3,25;"█*█*█*█"
 191 POKE 22783,149: POKE 22778,
149: POKE 22773,149: POKE 22768,
149: POKE 22763,149: POKE 22753,
149
 192 PRINT AT 2,12;"#######";AT
7,12;"####";AT 7,22;"####";AT 7,
27;"####";AT 8,0;" ######";AT 8,
```

```
10;"###";AT 8,15;"###";AT 8,21;"
#####";AT 8,27;"#####"
 193 PRINT AT 13,0;" ▛▜*▛▜*▛▜*▛
*";AT 13,16;"▛▜▛▜▛▜▛▜****▛▜▛▜**"
 194 PRINT AT 17,0;" ####";AT 17
,6;"###";AT 17,11;"▛▀▀▀▀▀▀▀▜";AT 17,
19;"##*";AT 17,24;"*##*##*#";AT
18,2;"####";AT 18,21;"####";AT 1
8,26;"####"
 195 PRINT INK 1;AT 2,6;"C";AT 1
6,13;"C";AT 1,30;"C": PRINT INK
0;AT 20,30;"C": REM GRAPHICS c
 196 POKE 23105,148: POKE 23110,
148: POKE 23124,148: POKE 23129,
148: POKE 23134,148
 205 FOR n=0 TO 21
 208 IF n<=15 THEN PRINT INK 2;A
T 5,2*n+1;"O": REM GRAPHICS q
 210 IF n<=14 THEN PRINT INK 4;A
T 5,2*n+3;"O": IF n<=16 THEN PRI
NT AT 5,2*n-2;"  ": PRINT AT 5,
29;"  "
 215 IF n<=17 THEN PRINT INK 1;A
T 4,2*n-4;"O": PRINT AT 4,2*n-6;
"  ": IF n>16 THEN PRINT AT 4,29
;"  "
 220 IF n<=15 THEN PRINT INK 2;A
T 10,2*n+1;"O": PRINT AT 10,2*n-
1;"  "
 225 IF n>1 AND n<=16 THEN POKE
22880+2*n-2,163: PRINT AT 11,2*n
-4;"  ": PRINT AT 11,29;"  "
 230 IF n<=14 THEN PRINT INK 3;A
T 15,28-2*n;"PRS": IF n<=15 THEN
 PRINT AT 15,31-2*n;"  ": REM G
RAPHICS prs
 240 IF n<=10 THEN PRINT INK 1;A
T 19,3*n+1;"O": PRINT AT 19,3*n-
2;"  "
 250 IF n<=19 THEN PRINT INK 3;A
T n+2,n+2;"O": PRINT AT n+1,n+1;
"  "
 255 IF n<=21 THEN PRINT INK 1;A
T n,26-n;"O": PRINT AT n-1,27-n;
"  "
 259 PRINT AT 15,0;"  "
 260 PRINT AT 21,0;"
 "
 261 PRINT AT 0,1;"
 ": PRINT AT 20,0
;"  ": PRINT AT 10,31;"  ": PRINT
AT 5,31;"  ": PRINT AT 19,31;"  ":
 PRINT AT 11,0;"  ": PRINT AT 15,
0;"  ": PRINT AT 6,0;"  "
 280 LET a=x: LET b=y
 285 IF INKEY$="z" AND x<21 THEN
LET x=x+1
 286 IF INKEY$="1" AND x>1 THEN
LET x=x-1
 290 IF INKEY$="m" AND y>1 THEN
LET y=y-1
 291 IF INKEY$="p" AND y<31 THEN
LET y=y+1
 300 PRINT AT a,b;"  "
 310 PRINT INK 5;AT x,y;"O"
 312 IF SCREEN$ (x,y-1)="*" THEN
LET s=s+5
 314 IF SCREEN$ (x,y+1)="*" THEN
LET s=s+12
 316 IF SCREEN$ (x,y-1)="#" THEN
LET s=s+10
 318 IF SCREEN$ (x,y+1)="#" THEN
LET s=s+12
 320 IF x=18 AND y=1 OR x=18 AND
y=6 OR x=18 AND y=20 OR x=18 AN
D y=25 OR x=18 AND y=30 THEN GO
TO 400
 322 IF x=7 AND y=1 OR x=7 AND y
=11 OR x=7 AND y=16 OR x=7 AND y
=21 OR x=7 AND y=26 OR x=7 AND y
=31 THEN GO TO 400
 324 IF x=3 AND y=15 OR x=8 AND
y=13 OR x=8 AND y=14 OR x=8 AND
y=26 THEN GO TO 400
 326 IF x=17 AND y=5 OR x=17 AND
y=22 OR x=17 AND y=23 OR x=20 A
ND y=28 THEN GO TO 400
 328 IF x=16 AND y=13 THEN LET s
=s+15
 330 IF x=1 AND y=30 THEN LET s=
s+15
 332 IF x=2 AND y=6 THEN LET s=s
+15
 335 IF x=4 AND y=2*n-4 OR x=4 A
ND y=2*n-3 THEN GO TO 400
 340 IF x=5 AND y=2*n+3 OR x=5 A
ND y=2*n+2 THEN GO TO 400
 344 IF x=10 AND y=2*n+1 OR x=10
AND y=2*n+2 THEN GO TO 400:
 346 IF x=15 AND y=29-2*n OR x=1
5 AND y=30-2*n THEN GO TO 400
 347 IF x=19 AND y=3*n+1 OR x=19
 AND y=3*n OR x=19 AND y=3*n-1 T
HEN GO TO 400
 348 IF x=n+2 AND y=n+2 THEN GO
TO 400
 349 IF x=n AND y=26-n THEN GO T
O 400
 350 IF x=20 AND y=30 THEN LET s
=s+100
 352 IF x=20 AND y=30 THEN GO TO
 380
 360 LET t=FN t()
 363 PRINT INK 0; PAPER 7;AT 0,2
9;t-t1;"  "
 365 IF t=t1+120 THEN GO TO 380
 366 NEXT n
 370 GO TO 205
 380 CLS
 383 IF s>hsc THEN LET hsc=s
 385 PRINT FLASH 1; INK 0; PAPER
 7;AT 5,11;"GAME OVER": PRINT AT
 15,3;"If you wish to play again
,      press key c"
 387 PRINT AT 10,7;"YOUR SCORE I
S ";  FLASH 1; INK 7; PAPER 0;s;"
 "
 388 PRINT AT 13,7;"HIGH SCORE I
S ";  FLASH 1; INK 0; PAPER 7;hsc
;"  "
 390 FOR e=1 TO 25 STEP 4
 391 BORDER INT (e/4)
 392 BEEP .5*(SIN (e*PI/2)),25
 393 BEEP .5*(COS (e*PI/2)),0
 396 IF INKEY$="c" THEN GO TO 15
0
 397 NEXT e
 399 GO TO 385
 400 CLS
 405 IF s>hsc THEN LET hsc=s
 415 PRINT INK 0; PAPER 7; FLASH
 1;AT 2,10;"COLLISION!"
 418 PRINT AT 10,7;"YOUR SCORE I
S ";  FLASH 1; INK 7; PAPER 0;s;"
 "
 419 PRINT AT 13,7;"HIGH SCORE I
S ";  FLASH 1; INK 0; PAPER 7;hsc
;"  "
 420 PRINT AT 19,3;"If you wish
to play again,      press key c"
 421 FOR d=1 TO 29 STEP 4
 422 BORDER INT (d/4)
 423 BEEP .5,2*(SIN (d*PI/2))
 424 BEEP .5,-2*(SIN (d*PI/2))
 430 IF INKEY$="c" THEN GO TO 15
0
 431 NEXT d
 435 GO TO 415
```

Should you make it safely to your destination, this is the message you will be greeted with. Note the high score facility.

```
GAME OVER

YOUR SCORE IS 100

HIGH SCORE IS 100

If you wish to play again,
press key c
```

Transyl-vanian tower

Searching out Count Dracula is no easy task, as Phil Garratt will testify.

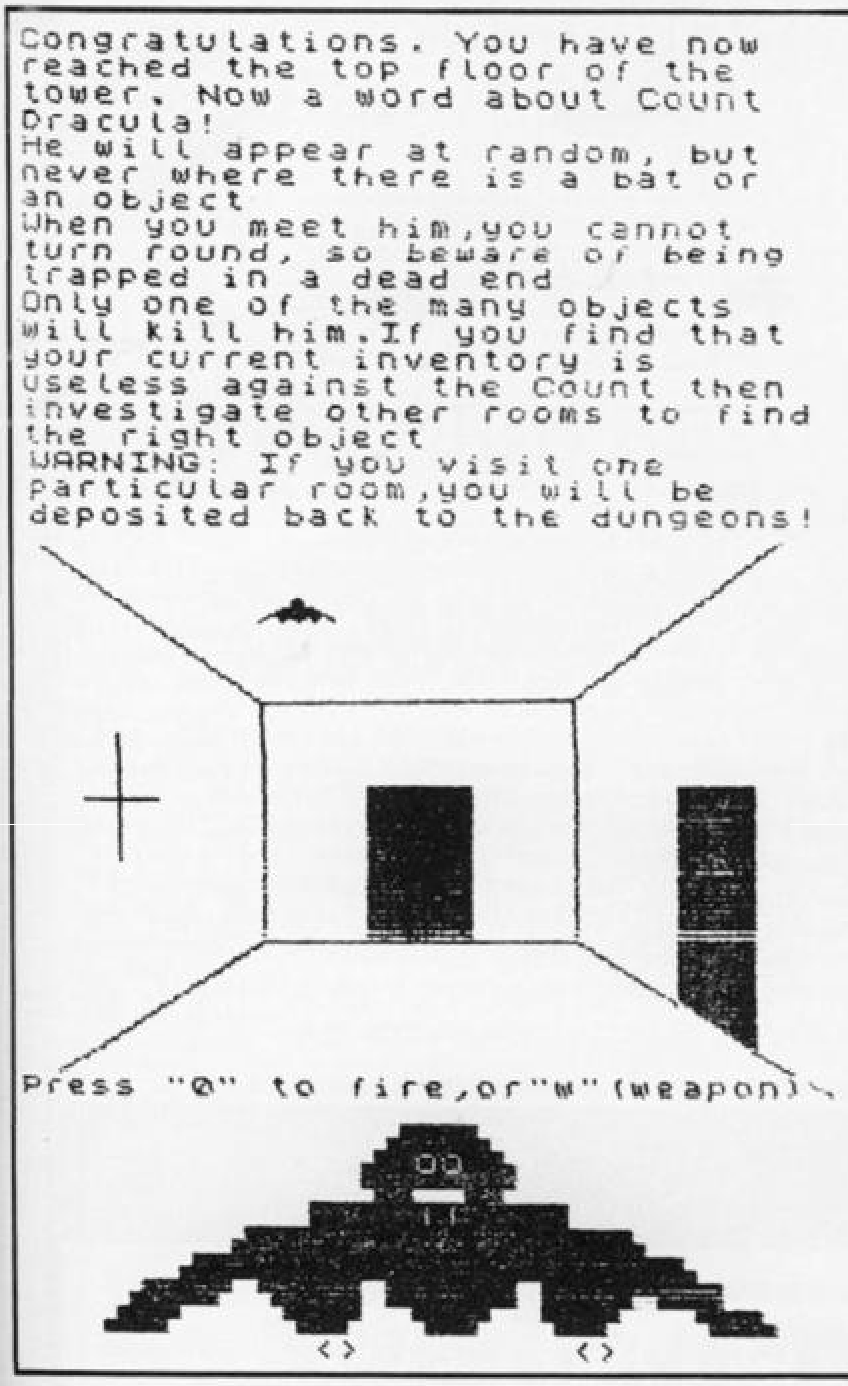
Transylvanian Tower is the latest of Richard Shepherd's adventure games for the 48K Spectrum. Count Dracula's lair contains no less than 500 rooms, and your task is to find your way through them, kill the Count, and steal from his secret treasure trove.

The tower is on five levels, each of which is made up of a ten by ten maze. You start in the top left corner, and you have to reach the room at the bottom right which contains the 'transporter' (what's wrong with stairs?!) to take you to the next level. As you enter each room, you are given a three-dimensional view of the walls and any doors leading off. You may also come across a useful object such as a ring that lets you walk through walls or a sword for killing bats with. Movement is done using the cursor keys, '5' and '8' for left and right, '7' to go forward, and '6' turns you 90 degrees anti-clockwise (so pressing it twice turns you right round without leaving the room).

Necking in the tower

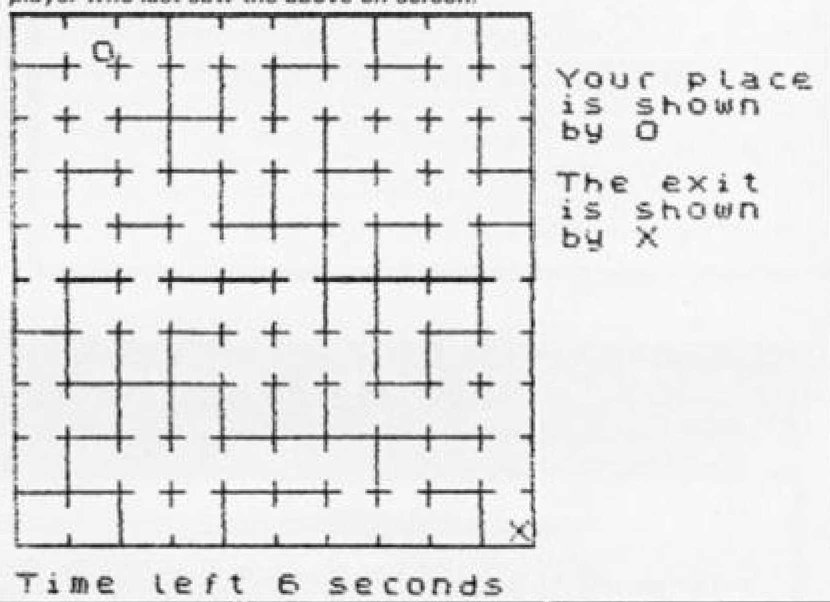
Once you are past the opening titles of the program, complete with sneak preview of the Count, you have to wait for over two minutes for the random maze to be set up. Your quest starts at level 1, which is the dungeon level, and contains neither hazards nor objects. Pressing 'H' at any time displays a floor plan, which is built up gradually line by line and takes over 30 seconds to complete. Your position and that of the exit are then shown, and after 10 seconds you are returned to the room display. Eventually, you reach the exit and you then have







Some sample screen illustrations from the program, Transylvanian tower. Instructions are provided along with graphic displays of some of the rooms you may visit. Below is a map of your progress around the Count's tower, but if you encounter the dreaded Count Dracula you'd better make sure you're a little more prepared than the unlucky player who last saw the above on-screen!



another two minutes plus wait while the second level maze is worked out.

You cross level 2 in the same way as level 1, but this time you have to cope with the vampire bats. These little monsters fly randomly around the ceiling of many of the rooms, and you either have to kill them or run away. Linger too long and the pain in the neck they give you is fatal! You have a gun to shoot them with, but only 10 bullets (although you may come across more scattered in the maze). When you fire, a rather messy line is drawn vertically up the middle of the screen, and you have to hit the bat with it. You cannot aim your shot, so you have to rely on the bat flying into the right position (or wrong position from the bat's point of view).

There are two other important things concerning bats. Firstly, you have to kill a certain number of them before you can progress to the next level (20 on the second; 30 on the third, and 40 on the fourth). Secondly, their radar system is a bit defective, so they all carry a floor plan so that they can find their way around the castle. This means that every time you kill a bat, you have a chance to find out where you are and where you want to go.

Also on level 2 and above are objects which might be of some use against the bats. There are eight other items apart from the ring and sword I mentioned earlier. There is a cloak that stops bats attacking, an apple that will speed you to the next level, a knife and a dagger, a floor plan, and finally three items without which no vampire movie would be complete: a mirror, a clove of garlic and a silver cross. You can only carry up to three objects at any one time though.

Levels 3 and 4 are like level 2, except that the bats are even harder to kill. If you reach level 5, then you get the chance to confront Count Dracula himself. Only one of the objects is an effective weapon against him, and I never lived long enough to find out which!

Fangs ain't what they used to be

Provided you have been playing the game for at least 30 minutes, you can save the game onto tape. The entire 36K of the program is saved rather than just the data. After using the SAVE feature, as soon as the current game finishes (win or lose) the program does a NEW and you have to load from tape to play again. The same applies to the game you just saved to tape.

I'm afraid I think that Richard Shepherd has a rather too high opinion of his own program if he expects many people to spend 3 minutes loading from tape, then 2 minutes waiting for the game to set itself up, only to have the program NEW if the SAVE feature is used.

Unlike Richard Shepherd's earlier adventures, Transylvanian Tower does not send any output to the printer. The program is entirely written in BASIC, but if you attempt to break the program, for example, to dump the floor plan to the printer, that also causes the program to NEW.

This is a reasonable game for the very patient, but would be greatly improved if the maze were set up by machine code. The graphics are the best so far from Richard Shepherd, but still only adequate rather than inspired. It also struck me that this would be a desperately hard game to win. If you do manage it, I suggest you treat yourself to a Bloody Mary!

Transylvanian Tower is priced at £6.50 from Richard Shepherd Software, Freepost, Maidenhead, Berks SL6 5BY.

ACTION PACKED
****

ARCADE STYLE
****

```
S  P  E  C  T  R  U  M
S  P  E  C  T  R  U  M
S  P  E  C  T  R  U  M
S  P  E  C  T  R  U  M
S  P  E  C  T  R  U  M
```

GAME

```
***********************************************************
*                                                         *
*  00   0    0 0000 00  0     0   0  0  00   0 0000       *
* 0  0  0    0 0    0 0 0     00 00  0  0 0  0 0          *
* 0000  0    0 0000 0  00     0 00 0 0  0  0 0 0000       *
* 0  0  0    0 0    0   00    0    0 0  0   00 0          *
* 0  0  000  0 0000 0   00    0    0 0  0   00 0000       *
*                                                         *
***********************************************************
```

This is a game of strategy, of forward planning. It is easy to play, but difficult to master. If you like a challenge you will love this. If you want the most from your Spectrum and yourself then try to survive in the 'alien mine'.

SUPERB HI-RES COLOUR GRAPHICS, AND EXCELLENT SOUND.

ONLY £4.99 INC VAT AND P&P.
RETURN OF POST MAIL ORDER SERVICE.

TO OBTAIN 'ALIEN MINE' SEND CHEQUE/PO's to
I.E. APPLICATIONS LTD. (DEPT ZXC)
177 CASTLETON ROAD,
MOTTINGHAM, LONDON.
SE9 4DE

(*** SPECTRUM PROGRAMS WANTED ***)

Spectrum 48k

MAKE YOUR OWN TV FILMS!



WITH LIVE ANIMATION

& MUSIC

new EXCITING!



SAVE them . . .

& AMUSE FRIENDS AND FAMILY WHEN YOU PLAY THEM BACK!



ORDER: £7 MACRO CONSTRUCTION & ANIMATION CASSETTE MCA

- SPECIAL MACRO-PROGRAMMING TECHNIQUES ALLOW YOU TO: -
- CALL-UP ANY MEMBER OF CAST, INCLUDING UFO, HELICOPTER AND A HOST OF 'PERSONNEL' IN A CHOICE OF NEARLY 30 'CALLS! MAKE EACH ANY COLOUR - LARGE ONES CAN BE IN TWO COLOURS!
- THEY WALK, RUN, CLIMB & FLY.
- BUILD HOUSES, CHURCHES ETC. IN A FLASH - & ERASE THEM
- UP TO 15 MINS, PER 'FILM', OR MORE IF YOU'RE A GOOD DIRECTOR
- MAKE CHARACTERS 'SPEAK' USING ANNOTATIONS AS IN STRIPS.
- FULL COMPUTER-PROMPTING TO SAVE YOU MAKING ERRORS.
- INSTANT PLAYBACK, EDITING AND DELETING AS YOU GO ALONG!
- MOST COMMANDS ARE A SINGLE KEY-STROKE, WITHOUT 'ENTER'.
- COMPOSE YOUR OWN TUNES USING THE SPECIAL THREE-OCTAVE CHROMATIC MUSIC COMPOSER, WHICH ALLOWS YOU TO LISTEN, CORRECT AND EDIT AT EACH NOTE ... IF YOU NEED TO!
- STORE UP TO NINE 99-NOTE TUNES TO 'CALL-UP' AT ANY TIME IN A STORY ... COMPUTER GIVES EACH A NUMBER IN MEMORY.
- CHANGE TUNES AT ANY TIME, EVEN WHILST MAKING A 'FILM'.
- YOU CAN EVEN ADD PERSPECTIVE TO BUILDINGS, ROADS ETC.
- NO SPECIAL SKILLS OR COMPUTER KNOWLEDGE NEEDED . . . JUST IMAGINATION!

pinehurst data studios
IMPORTANT: FOR 48K SPECTRUM ONLY

69, PINEHURST PARK,
WEST MOORS, WIMBORNE
DORSET. BH22 0BP

SPECTRUM 16/48

AWARI

★ The ancient African game of logic. It takes 2 minutes to learn the rules but far longer to master the tactics.

★ Select the 'Goat-herd' level of play and it's an addictive game for children (8+) that exercises their minds – not their laser fingers.

★ Select the 'Witch-doctor' level and it's a threat to your sanity. We haven't beaten it and we wrote it!

ZX81 16K

ADVENTURE

★ Over 200 places to explore in this machine code game using advanced data compression techniques.

★ No random elements – you will need skill, cunning and a sense of humour as you explore caves, forest and castles.

★ Evade ruthless pursuers and overcome a host of obstacles.

★ Multiple word commands and single letter abbreviations!

£1000 IN PRIZES

FANTASTIC VOYAGE (ZX81 16K ONLY)

This real-time graphics simulation set inside the human body was written by a lecturer in anatomy. You are injected into the blood stream in your miniature submarine. Navigate the arteries, veins and heart to the brain, where a blood clot must be destroyed. Features a real vascular map. You will be attacked by lymphocytes which must be destroyed using arcade game type graphics. Everything you do uses up precious energy. Three display formats – a lateral and frontal body scan plus blood vessel name, a close-up scan and a laser-sight for fighting lymphocytes.

★★ Buy both Awari and Adventure and enter the 'Foilkade Challenge' competition. Details with cassette or send SAE.

ALL GAMES £5.95 EACH, 2 FOR £9.95, 3 FOR £13.95 (ANY MIX) INCLUSIVE

FOILKADE LTD

DEPT PCT3
66 LITTLEDEAN, YATE
BRISTOL BS17 4UQ

Daredevil

Dice with death in this program for your 16K ZX81, courtesy of Mr T Jane of Crossways.



How do you fancy your chances of making a death defying leap, Evel Kneival style?

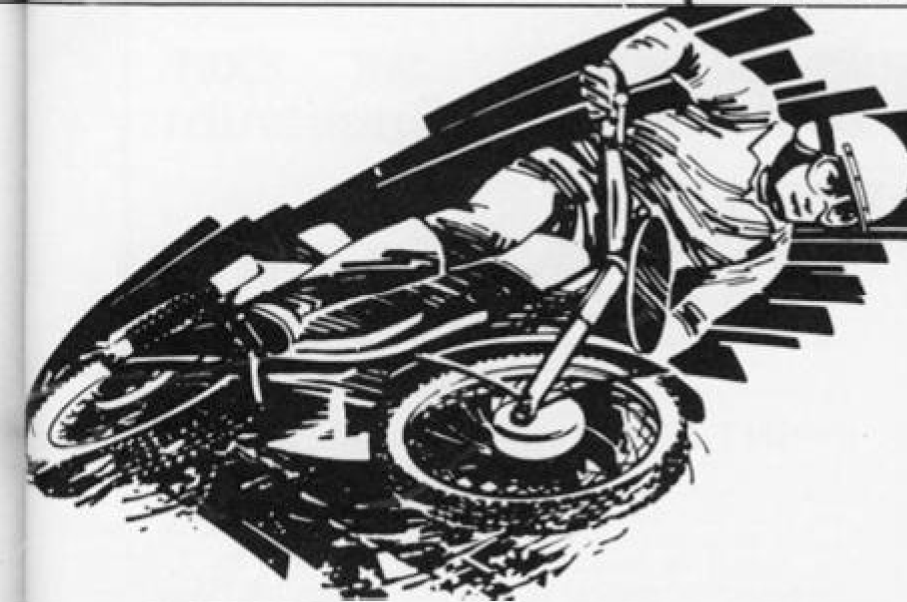
On RUNning the program, you are first asked if you would like to see instructions on how to play the game. Once you are ready to roll, the computer sets up a ramp (comprising a '/' followed by a Graphics 5 character) followed by a number of obstacles (made up from '$' signs).

Over the top

You are then invited to select the speed at which you wish to attempt the jump off the ramp. Once the speed has been input, the 'bike' is seen to rev up and then speed up the ramp; make a graceful leap (in the form of a parabolic curve) and land. Hopefully, you will have chosen a speed suitable to ensure that you get your bike over the obstacles.

To make life more demanding, there is also a safety wall just beyond the end of the obstacles and the 'bike' must land before it.

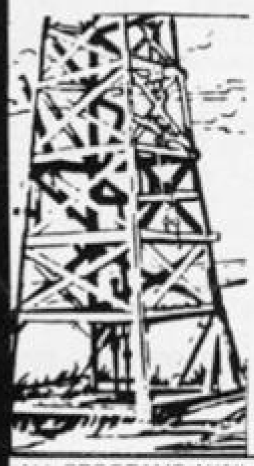
Full on-screen scoring of successful jumps and crashes is given, and after 10 jumps the ZX81 delivers a scathing report assessing your skill as a daredevil motor bike rider.



```
   1 PRINT "DO YOU WANT INSTRUCT
IONS? (Y/N)"
   2 INPUT A$
   3 IF A$="Y" THEN GOTO 3000
   4 IF A$<>"N" THEN GOTO 2
   5 LET CR=0
   6 LET SA=0
   7 CLS
   9 PRINT AT 0,6;"DARE DEVIL"
  10 PRINT AT 3,2;"CRASHES: ";CR;
AT 3,17;"GOOD JUMPS: ";SA
  12 IF CR+SA=10 THEN GOTO 4000
  14 FOR N=0 TO 31
  15 PRINT AT 17,N;"▒"
  20 NEXT N
  30 LET X=INT (RND*18)+5
  40 PRINT AT 16,5;"/█";AT 16,0;
"■"
  45 FOR N=7 TO 30-X
  50 PRINT AT 16,N;"$"
  55 NEXT N
  56 LET K=(N+4)*2
  57 PLOT K,10
  58 PLOT K,11
  60 GOSUB 2000
  70 FOR M=1 TO 20
  80 PRINT AT 15,0;"BRRRM"
  90 PRINT AT 15,0;"BRRRM"
 100 NEXT M
 110 PRINT AT 15,0;"     "
1000 FOR X=1 TO 9
1010 PLOT X,10
1020 UNPLOT X,10
1030 NEXT X
1050 FOR A=1 TO S
1060 LET B=A*PI/S
1070 PLOT A+11,SIN (B)*20+10
1080 UNPLOT A+11,SIN (B)*20+10
1090 NEXT A
1095 PLOT A+11,SIN (B)*20+10
1100 IF A+11>N*2 AND A+11<K THEN
 PRINT AT 21,20;"SAFE"
1105 IF A+11>N*2 AND A+11<K THEN
 LET SA=SA+1
1106 IF A+11>N*2 AND A+11<K THEN
 PAUSE 150
1110 IF A+11>N*2 AND A+11<K THEN
 GOTO 7
1120 PRINT AT 21,20;"CRASH"
1130 LET CR=CR+1
1135 PAUSE 150
1140 GOTO 7
2000 REM ACCEPT THROTTLE AND CAL
CULATE DISTANCE
2005 PRINT AT 21,0;"SELECT SPEED
 (20 - 80 MPH)"
2010 INPUT V
2011 PRINT AT 21,0;V;" MPH
 "
2015 IF V>80 THEN LET V=80
2016 IF V<20 THEN PRINT AT 21,0;
"TOO SLOW"
2017 IF V<20 THEN PAUSE 100
2018 IF V<20 THEN GOTO 2005
2020 LET U=(V*1000)/3600
2030 LET A=90/180*PI
2040 LET S=INT (((U**2)*SIN (A))
/9.81)
2050 RETURN
3000 REM INSTRUCTIONS
3005 CLS
3010 PRINT "           DARE DEVIL
"
3018 PRINT
3020 PRINT "YOU ARE GOING TO PER
FORM"
3030 PRINT "A DEATH DEFYING LEAP
ON A"
3040 PRINT "MOTORBIKE OVER A"
3050 PRINT "SET OF OBSTACLES.THE
RE IS A"
3060 PRINT "RAMP MARKED /█,AND T
HE OBSTACLES"
3070 PRINT "ARE INDICATED BY""$"
""
3080 PRINT
3090 PRINT "YOU CAN SELECT ANY S
PEED UP"
3100 PRINT "TO 80MPH,BUT BEWARE
IF"
3110 PRINT "YOU JUMP TOO FAR YOU
 WILL "
3120 PRINT "CRASH INTO THE SAFET
Y WALL."
3125 PRINT "YOU WILL NOT MAKE IT
 UP THE ","RAMP IF YOU GO LESS T
HAN 20 MPH"
3130 PRINT
3131 PRINT "YOU WILL GET 10 ATTE
MPTS."
3132 PRINT "GOOD LUCK."
3133 PRINT
3140 PRINT "PRESS NEWLINE TO CON
TINUE"
3150 INPUT I$
3160 GOTO 5
4000 REM  END GAME
4010 PRINT AT 5,9;"RUNS COMPLETE
";AT 7,7;"ASSESSMENT FOLLOWS"
4015 PAUSE 100
4020 IF SA<=1 THEN PRINT AT 9,0;
"YOU ARE A DANGER TO SOCIETY ON
A BIKE"
4025 IF SA=2 THEN PRINT AT 9,0;"
COME ON NOW,YOU CAN DO BETTER
THAN THAT."
4030 IF SA=3 THEN PRINT AT 9,0;"
YOU HAVE A LONG WAY TO GO BUT","
THE CROWD APPLAUD YOU FOR","TRYI
NG."
4035 IF SA=4 THEN PRINT AT 9,0;"
NOT BAD BUT NOBODY WANTS YOUR" "
AUTOGRAPH YET."
4040 IF SA=5 THEN PRINT AT 9,0;"
COMING ON,THE CROWD LIKED YOUR",
"STYLE."
4045 IF SA=6 THEN PRINT AT 9,0;"
PRETTY GOOD. YOU,LL MAKE THE ","
TOP YET."
4050 IF SA=7 THEN PRINT AT 9,0;"
VERY SKILLFUL,THE FANS LOVE YOU"
4055 IF SA=8 THEN PRINT AT 9,0;"
EXCELLENT RIDING,WELL DONE YOU",
"ARE IN THE TOP TEN RIDERS."
4060 IF SA=9 THEN PRINT AT 9,0;"
SUPERB RIDING. YOU ARE A HERO."
4065 IF SA=10 THEN PRINT AT 9,0;
"ABSOLUTELY OUTSTANDING. YOU","A
RE THE BEST"
4070 PRINT AT 12,0;"ANOTHER GAME
 (Y/N) ?"
4075 IF INKEY$="" THEN GOTO 4075
4076 IF INKEY$="Y" THEN GOTO 5
4080 IF INKEY$<>"N" THEN GOTO 40
75
4090 PRINT AT 17,8;"OK BYE FOR N
OW"
```

MANAGEMENT GAMES

16K ZX81
Spectrum



DALLAS

A game of oil exploration and exploitation in Texas. Decisions are required on purchasing concessions, the movement and use of drilling rigs, and the building of platforms and pipelines. Can you take-over the Ewing Empire, or will you be taken-over in the attempt!

AUTOCHEF

As MD of a Catering Company you must negotiate for leases, decide on menu prices, level of wages, advertising and dividends. Each year you must predict the inflation rate. You will be made to resign if you are not successful. There are 3 levels of difficulty.

AIRLINE

Can you compete with British Airways? You must decide on the number of aircraft to operate, whether to buy or charter, and the levels of staffing and maintenance. Problems encountered are tax demands, strikes, cancelled flights, hijacks and aircraft crashes.

PRINT SHOP

You own a small printing company and are required to decide on (a) the number and type of staff you employ and when to increase or reduce staff (b) the amount and type of paper you stock (c) the week in which work is scheduled (d) the quotation for each. There are 3 scales of difficulty.



ALL PROGRAMS AVAILABLE FOR 16K ZX-81 SPECTRUM 16K AND 48K

NEW **FARMER** is a simulation of running a large wheat farm. Decisions are required on when to plant, irrigate, fertilize, spray and harvest your land and whether to purchase seed corn, land or farm machinery or whether to employ more farm workers.

EXCELLENT ROYALTIES PAID FOR SINCLAIR SOFTWARE

PRICES TO ORDER **(MAIL ORDER)** ZX81 16K & SPECTRUM 16K, £5 for 1, £8.50 for 2, £12 for 3, £15.50 for 4 & £19 for 5.
SPECTRUM 48K, £6 for 1, £10.50 for 2, £15 for 3, £19.50 for 4 & £24 for 5.
Please state computer type and send cheque to:-

DEPS.S., CASES COMPUTER SIMULATIONS, 14 LANGTON WAY, LONDON SE3 7TL.

RETAIL STOCKISTS

LONDON	Buffer Micro Shop, 310 Streatham High Road, S.W.16.
	Data-Assette, 44 Shroton Street, N.W.1.
	The Games Centre, Oxford Street & Regent Street, W.1.
BATH	Micro Style, 29 Belvedere, Lansdown Road
BOURNEMOUTH	Zedextra, 5 School Lane, Kinson
BRIGHTON	Gamer, 25 Gloucester Road
DAVENTRY	Micro Style, 67 High Street
IPSWICH	Brainwave Micros Ltd., 24 Crown Street
LEEDS	Dragon Byte, 51a Queen Street, Morley
LEICESTER	Microware, Unit 5, St. Peters Lane
MANCHESTER	J.M. Computers, 136 Park Lane, Whitefield
	Micro-Link, 830 Hyde Road
NEWBURY	Micro Style, 47 Cheap Street
NEW ELTHAM	Kayde Home Computers, 1 Station Approach
NORFOLK	Kayde Electronics Ltd., The Conge, Great Yarmouth

NOW AVAILABLE AT W. H. SMITH

NEW

SAVE IT!

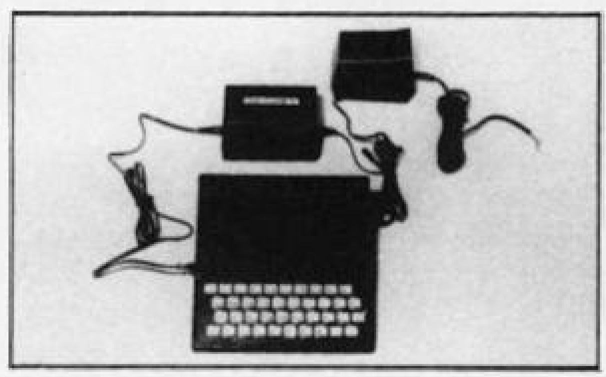
FOR ZX 81 AND SPECTRUM COMPUTERS

(AND OTHER MODELS USING 9v DC ADAPTORS)

with battpack©

(Patent applied for)

Only **£13·95** + £1·40 p/p



PROTECT YOUR PROGRAMMES WITH THIS RECHARGEABLE BATTERY PACK

AT LAST! The answer to one of the major problems facing personal computer programmers and users; — **POOR MAINS SUPPLIES CAUSING HAVOC, WHILE YOU SPEND HOURS TRYING TO ENTER YOUR CAREFULLY PREPARED PROGRAMMES.**

Now Adaptors and Eliminators, who have produced nearly 70% of all ZX Mains Adaptors, bring you **BATTPACK** — a rechargable 9v Ni-Cad battery pack and transient suppressor, housed in a complementary black ABS case that simply plugs into your ZX adaptor in seconds.

BATTPACK ensures the voltage into your ZX computer never drops below 9v, thereby eliminating the possibility of a programme crash because of a split second mains dropout! (One cycle is all it needs!)

BATTPACK also suppresses transients and RFI using carefully selected capacitors. Your ZX Adaptor keeps the **BATTPACK** at full charge even when in use — so its always ready for emergencies. Even in the event of a full power cut **BATTPACK** will run your ZX computer for at least 30 minutes giving you time to save your programme on a cassette or print a hard copy on your ZX printer.

BATTPACK comes complete with all connecting leads and is ready for use.

ORDER NOW — SAVE NOT ONLY YOUR PROGRAMMES BUT ALSO YOUR TEMPER!!

MAIL ORDER ONLY. PLEASE ALLOW 28 DAYS FOR DELIVERY.

ADAPTORS AND ELIMINATORS LTD. 14 THAMES ST. LOUTH. LINCS.

PLEASE SUPPLY______ (QTY) battpack/s
AT £13·95 each PLUS £1·40 p/p
I ENCLOSE CHEQUE POSTAL ORDER FOR £______

NAME
ADDRESS

MACHINE SPECIFICATIONS

ZX80

Dimensions
Width 174mm (6.85 in)
Depth 218mm (8.58 in)
Height 38 mm (1. 5 in)
Weight 300g (10.5oz)

Microprocessor/Memory
Z80A 3.25 MHz clock
ROM: 4K bytes containing BASIC
RAM: 1K bytes internal, externally expandable to 16K bytes.

Display
Requires an ordinary domestic black and white colour TV. The lead supplied connects between the ZX80 and your TV's aerial socket. The display organisation is 24 lines of 32 characters per line showing black characters on a white screen. The ZX80 does not connect to a printer.

Programming
Programs can be entered on the keyboard or loaded from cassette. The ZX80 has automatic "wrap round" so lines of program can be any length but not multi-statement lines.

Syntax check
The syntax of the entered line is checked character by character. A syntax error cursor marks the first place the syntax breaks down if there is an error. Once any errors have been edited out the syntax error cursor disappears. Only syntax error-free lines of code are accepted by the ZX80.

Graphics
Total of 22 graphics symbols giving 48 x 64 pixels resolution consisting of 10 symbols plus space and inverses. Includes symbols for drawing bar charts. Under control of your BASIC program any character can be printed in reverse field.

Editing
The line edit allows you to edit any line of program or input including statement numbers. The edit and cursor control keys are EDIT, RUBOUT, HOME.

Arithmetic
Arithmetic operators +,–,x, ÷ exponentiate. Relational operators <, >, = , yielding 0 or – 1. Logical operators AND OR NOT yielding boolean result. Relational operators also apply to strings. ZX80 BASIC uses 16 bit two's complement arithmetic (± 32767).

Variables
Numeric variable names may be any length, must begin with a letter and consist of alphanumerics. Every character in the name is compared thus an infinity of unique names is available.

String variables may be assigned to or from, shortened but not concatenated. String variable names are A$ – Z$. Strings do not require a dimension statement and can be any length.

Arrays have a maximum dimension of 255 (256 elements) each. Array names consist of a single letter A–Z.

Control variable names in FOR. . . NEXT loops consist of a single letter A–Z.

Expression evaluator
The full expression evaluator is called whenever a constant or variable is encountered during program execution. This allows you to use expressions in place of constants especially useful in GOTOs, GOSUBs, FOR. . . NEXT etc.

Immediate mode
The ZX80 will function in the "calculator mode" by immediately executing a statement if it is not preceded with a line number.

Cassette interface
Works with most domestic cassette recorders. The transfer rate is 250 baud using a unique tape-recording format. Other systems are not compatible with the ZX80's. The ZX80 also SAVEs the variables as well as the program on cassette. Therefore you can save the data for updating next time the program is executed. The ZX80 does not support separate data files. The lead supplied with the ZX80 is fitted with 3.5mm jack plugs.

Expansion bus
At the rear has 8 data, 16 address, 13 control lines from the processor and Ov, 5v, 9-11v, Ø and internal memory control line. These signals enable you to interface the ZX80 to your own electronics, PIO, CTC, SIO if you want I/O ports etc.

Power supply
The ZX80 requires approximately 400mA from 7–11v DC. It has its own internal 5v regulator.

TV standard
The ZX80 is designed to work with UHF TVs (channel 36)and is the version required for use in the United Kingdom. The ZX80 USA is designed to work with a VHF TV(American channel 2. European channel 3) and is the version required for the American TV system, also for countries without UHF.

ZX81

Dimensions
Width 167mm (6.32 in)
Depth 175mm (6.80 in)
Height 40 mm (1.57 in)
Weight 350 gms (12.15 oz)

Microprocessor/Memory
Z80A 3.25 MHz clock
ROM: Containing 8K BASIC interpreter
RAM: 1K bytes internal, externally expandable to 16K bytes.

Keyboard
40 key touch-sensitive membrane. Using function mode and single press key-word system, this gives the equivalent of 91 keys and also graphics mode allows an additional 20 graphical and 54 inverse video characters to be entered directly.

Display
Requires an ordinary domestic black and white or colour TV. The aerial lead supplied connects the ZX81 to the TV aerial socket. The display is organised as 24 lines of 32 characters with black characters on a white background.

Two mode speeds
The ZX81 can operate in two software-selectable modes - FAST and NORMAL. FAST is ideal for really high-speed computing. In NORMAL mode however the ZX81 allows continuously moving, flicker-free animated displays.

Printer
The 8K ROM will permit instructions (LPRINT, LLIST and COPY) to drive the Sinclair ZX Printer.

Programming
Programs can be entered via the keyboard or loaded from cassette. Programs and data can be saved onto cassette so that they

are not lost when the ZX81 is turned off.
Syntax check
The syntax of a line of program is checked on entry. A syntax error cursor marks the first place the syntax breaks down if there is an error. The syntax error cursor disappears when errors have been corrected. Only lines free from syntax errors will be entered into the program.
Graphics
Apart from the 20 graphics characters, space and its inverse, the display may also be divided into 64 x 44 pixels, each of which may be 'blacked' in or 'whited' out under program control.
Editing
A line editor allows you to edit any line of program or input, including program line numbers. Lines may be deleted, increased or decreased in size.
Arithmetic
Arithmetic operators +, –, x, ÷, exponentiate. Relational operators =, < >, >, <,< =,>=, may compare string and arithmetic variables to yeild 0 (False) or 1(True). Logical operators AND, OR, NOT yield boolean results.
Floating-point numbers
Numbers are stored in 5 bytes in floating-point binary form giving a range of $\pm$ 3 x 10^{-39} to $\pm$ 7 x 10^{38} accurate to 9½ decimal digits.
Scientific functions
Natural logs/antilogs; SIN, COS, TAN and their inverses;SQR; e^x.
Variables

Numerical:	any letter followed by alphanumerics
String:	A$ to Z$
FOR-NEXT loops:	A–Z (loops may be nested to any depth.
Numerical arrays:	A–Z
String arrays:	A$ to Z$

Arrays
Arrays may be multi-dimensional with subscripts starting at 1.

Expression evaluator
The full expression evaluator is called whenever an expression, constant or variable is encountered during program execution. This powerful feature allows use of expressions in place of constants and is especially useful in GOTO, GOSUB etc.
Command mode
The ZX81 will execute statements immediately, enabling it to perform like a calculator.
Cassette interface
Works using domestic cassette recorders. The transfer rate is 250 baud and uses a unique recording format not compatible with other systems. The ZX81 will save the data as well as the program to avoid the need to re-enter the data when the program is next loaded.,
ZX81 will search through a tape for the required program). The cassette leads supplied have 3.5 mm jack plugs.
Expansion port
At the rear, this has the full data, address and control buses from the Z80A CPU as well as OV, +5V, +9V, $\overline{\emptyset}$ and the memory select lines. These signals enable you to interface the ZX81 to the Sinclair 16K RAM pack and ZX printer.
Power supply
The ZX81 requires approximately 420mA at 7–11V DC. It has its own internal 5V regulator. The ready assembled ZX81 comes complete with a power supply. The ZX81 kit does not include a power supply.

TV standard
The ZX81 is designed to work with UHF TVs (channel 36) 625 lines.

ZX SPECTRUM

Dimensions

Width 233 mm
Depth 144 mm
Height 30 mm

CPU/Memory

Z80A microprocessor running at 3.5 MHz. 16K-byte ROM containing BASIC interpreter and operating system.
16K-byte RAM (plus optional 32K-byte RAM on internal expansion board) or 48K-byte RAM.

Keyboard

40-key keyboard with upper and lower case with capitals lock feature. All BASIC words obtained by single keys, plus 16 graphics characters, 22 colour control codes and 21 user-definable graphics characters. All keys have auto repeat.

Display

Memory-mapped display of 256 pixels x 192 pixels; plus one attributes byte per character square, defining one of eight foreground colours, one of eight background colours, normal or extra brightness and flashing or steady. Screen border colour also settable to one of eight colours. Will drive a PAL UHF colour TV set, or black and white set (which will give a scale of grey), on channel 36.

Sound

Internal loudspeaker can be operated over more than 10 octaves (actually 130 semitones) via basic BEEP command. Jack sockets at the rear of computer allow connections to external amplifier/speaker.

Graphics

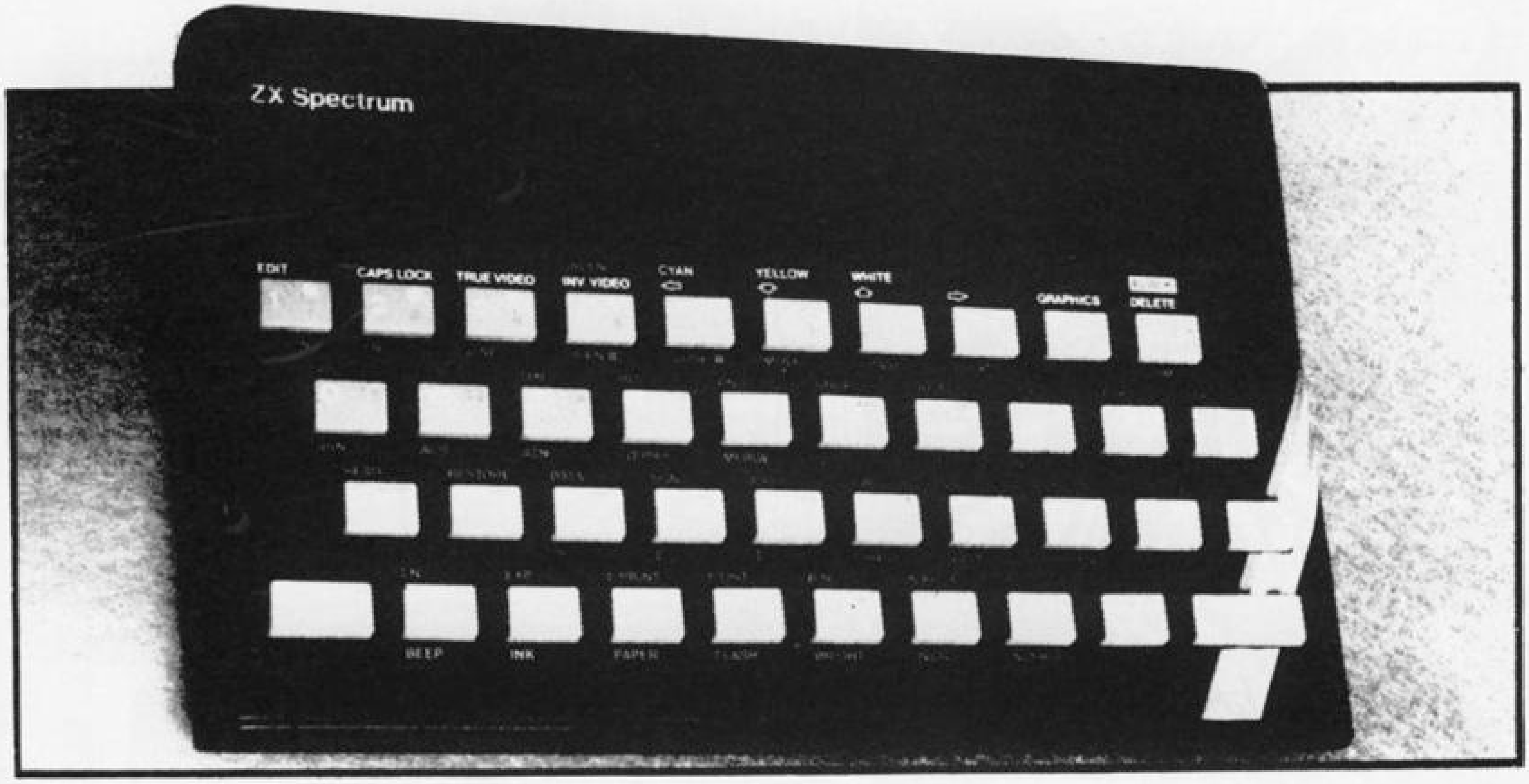
Point, line, circle and arc drawing commands in high-resolution graphics.
16 pre-defined graphics characters plus 21 user-definable graphics characters. Also functions to yield character at a given position, attribute at a given position (colours, brightness and flash) and whether a given pixel is set. Text may be written on the screen on 24 lines of 32 characters. Text and graphics may be freely mixed.

Colours

Foreground and background colours, brightness and flashing are set by BASIC INK, PAPER, BRIGHT and FLASH commands. OVER may also be set, which performs an exclusive — or operation to overwrite any printing or plotting that is already on the screen. INVERSE will give inverse video printing. These six commands may be set globally to cover all further PRINT, PLOT, DRAW or CIRCLE commands, or locally within these commands to cover only the results of that command. They may also be set locally to cover text printed by an INPUT statement. Colour-control codes, which may be accessed from the keyboard, may be inserted into text or program listing, and when displayed will override the globally set colours until another control code is encountered. Brightness and flashing codes may be inserted into program or text, similarly. Colour-control codes in a program listing have no effect on its execution. Border colour is set by a BORDER command. The eight colours available are black, blue, red, magneta, green, cyan, yellow and white. All eight colours may be present on the screen at once, with some areas flashing and others steady, and any area may be highlighted extra bright.

Screen

The screen is divided into two sections. The top section — normally the first 22 lines — displays the program listing or the results of program or command execution. The bottom section — normally the last 2 lines — shows the command or program line currently being entered, or the program line currently being edited. It also shows the report messages. Full editing facilities of cursor left, cursor right, insert and delete (with auto-repeat facility) are available over this line. The bottom section will expand to accept a current line of up to 22 lines.



Mathematical Operations And Functions

Arithmetic operations of +, −, ×, ÷, and raise to a power. Mathematical functions of sine, cosine, tangent and their inverses; natural logs and exponentials; sign function, absolute value function, and integer function; square root function, random number generation, and pi.
Numbers are stored as five bytes of floating point binary — giving a range of $+3 \times 10^{-39}$ to $+7 \times 10^{38}$ accurate to 9½ decimal digits.
Binary numbers may be entered directly with the BIN function. =, >, <, >=, <= and <> may be used to compare string or arithmetic values or variables to yield 0 (false) or 1 (true). Logical operators AND, OR and NOT yield boolean results but will accept 0 (false) and any number (true).
User-definable functions are defined using DEF FN, and called using FN. They may take up to 26 numeric and 26 string arguments, and may yield string or numeric results.
There is a full DATA mechanism, using the commands READ, DATA and RESTORE.
A real-time clock is obtainable.

String Operations And Functions

Strings can be concatenated with +. String variables or values may be compared with =, >, <, >=, <=, <> to give boolean results. String functions are VAL, VAL$, STR$ and LEN. CHR$ and CODE convert numbers to characters and vice versa, using the ASCII code. A string slicing mechanism exists, using the form a$ (x TO y).

Variable Names

Numeric — any string starting with a letter (upper and lower case are not distinguished between, and spaces are ignored).
String — A$ to Z$.
FOR-NEXT loops — A-Z.
Numeric arrays — A-Z.
String arrays — A$ to Z$.
Simple variables and arrays with the same name are allowed and distinguished between.

Arrays

Arrays may be multi-dimensional, with subscripts starting at 1. String arrays, technically character arrays, may have their last subscript omitted, yielding a string.

Expression Evaluator

A full expression evaluator is called during program execution whenever an expression, constant or variable is encountered. This allows the use of expressions as arguments to GOTO, GOSUB, etc.
It also operates on commands allowing the ZX Spectrum to operate as a calculator.

Cassette Interface

A tone leader is recorded before the information to overcome the automatic recording level fluctuations of some tape recorders, and a Schmitt trigger is used to remove noise on playback.
All saved information is started with a header containing information as to its type, title, length and address information. Program, screens, blocks of memory, string and character arrays may all be saved separately.
Programs, blocks of memory and arrays may be verified after saving.
Programs and arrays may be merged from tape to combine them with the existing contents of memory. Where two line numbers or variables names coincide, the old one is overwritten.
Programs may be saved with a line number, where execution will start immediately on loading.
The cassette interface runs at 1500 baud, through two 3.5 mm jack plugs.

Expansion Port

This has the full data, address and control busses from the Z80A, and is used to interface to the ZX Printer, the RS232 and NET interfaces and the ZX Microdrives. IN and OUT commands give the I/O port equivalents of PEEK and POKE.

ZX81 Compatibility

ZX81 BASIC is essentially a subset of ZX Spectrum BASIC. The differences are as follows.
FAST and SLOW: the ZX Spectrum operates at the speed of the ZX81 in FAST mode with the steady display of SLOW mode, and does not include these commands.
SCROLL: the ZX Spectrum scrolls automatically, asking the operator "scroll?" every time a screen is filled.
UNPLOT: the ZX Spectrum can unplot a pixel using PLOT OVER, and thus achieves unplot.
Character set: the ZX Spectrum uses the ASCII character set, as opposed to the ZX81 non-standard set.

ADVERTISERS INDEX

ZX COMPUTING

Lineage: 35p per word.
Semi display: £8.00 per single column centimetre
Ring for information on series bookings/discounts.

All advertisements in this section must be prepaid.
Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

01-437 1002
EXT 282
Send your requirements to:
SALLY COLLINS
ASP LTD, 145 Charing Cross Road,
London WC2H 0EE.

SOFTWARE EDUCATIONAL

Real Educational Software for the Spectrum
INKOSI — adventure/simulation Be an African King! **£5.95**
INVISIBLE MAN — Co-ordinates, compass points **£5.95**
PUNC-MAN — he gobbles the punctuation marks!

WANTED: Freelance programmers to convert existing BBC software to Spectrum. Good rates of pay.

WANTED Quality educational software. We publish, you get the royalties.

chalksoft — Send SAE to: **CHALKSOFT, Lowmoor Cottage, Tonedale, Wellington, Sometset TA21 0AL. Tel: 082 347 7117**

TEACH YOUR CHILD



to **TELL THE TIME** and **COUNT MONEY**
Two Fun Graphics Programs for **Spectrum (16K)** on one cassette for **£5.50** inc p&p from:

POPPY PROGRAMS, Richmond House, Ingleton, Carnforth LA6 3AN.

JUNIOR SCHOOL TUTOR
16K Spectrum
Fun for all ages from 4-44 years
FANTASTIC VALUE FOR MONEY!!
7 Educational and stimulating programs on one cassette.
• **Arithmetic** — 6 levels of play
• **Drawing** • **Spelling** • **Music Teacher**
• **Numbers** • **Snap** • **Keyboard Trainer**
Written by teachers. Suitable home or school.
ALL 7 PROGRAMS FOR JUST £4.95!
Send cheque or PO to:
ESSEX SOFTWARE
16 Huntsmans Drive, Upminster, Essex

Advertise your Educational Software in ZX COMPUTING for the best results.

ASP CLASSIFIED
01-437 1002 Ext. 282

PHYSICS
G.C.E. 'O' LEVEL
A.E.B. SYLLABUS 052 1983/4
16/48K SPECTRUM
COMPREHENSIVE EXAMINATION REVISION INCLUDING TUITION
£22
Program and details from:
HOMESTUDY LTD.
TRELEIGH WOODS FARM
REDRUTH, CORNWALL.

SOFTWARE APPLICATIONS

SPY CASSETTE. Amazing cassette will allow you to stop, list or copy any previously unstoppable tape. Includes two free utilities. State if for ZX81 or Spectrum. Send just £3. Bobker, 29 Chadderton Drive, Unsworth, Bury, Lancs.

ZX81-16K Adult users only. Free complete personal financial analysis program. Telephone Upper Warlingham 2280 (Ansaphone).

ZX Money Programs quickly and concisely calculates Savings, Pensions, Credit, energy in ZX Basic. Send £2.00. ZX Money, 28 Leygreen Close, Luton LU2 LOQ.

FIREWORK MUSIC
An educational game for the 16K or 48K Spectrum designed to develop speed in music reading. Players learn to read notes in the treble and the base clef, while they enjoy an attractive game with colour graphics and sound. There are firework displays as rewards but your firework is blown up by falling sparks if you're too slow. Pace increases with score to encourage quick reading. Ideal for home or school use by music beginners. 2 programs on cassette for 16K or 48K Spectrum (or a similar 16K ZX81 version) cost only **£5.00**. Also available, 3 Music Education Software cassettes for 16K ZX81, just **£12.00**
SOFTWARE COTTAGE
19 Westfield Drive, Loughborough, Leics LE11 3QJ.

Keep the Bank Manager at bay with . . . Homac
ZX Home Accounting System
— Database of up to 1,000 records Spectrum: 3,000 for book-keeping and financial forecasting.
— Fast data entry.
— Manages most accounts: bank, credit card, cash, savings, standing orders, direct debits etc.
— Predicts credit card bills.
— Up to 94 account headings covering an eighteen month period (Spectrum: 30 months) with annual carry forward.
— Printed or displayed summaries and statements for any account or period.
— Spectrum version provides for up to twenty currencies or individuals; ZX81 for any single currency only.

HOMAC £8.95
Please state if ZX81 (16K) or Spectrum (48K) version is required. Prices include cassettes and instructions. Post and packing **£1.05**.
David Martin Associates
71A Princes Road, Richmond, Surrey TW10 6DQ.
Telephone: 01-948 1053

SPECTRUM GRAPH PLOTTER
Draws pie-charts, bar-charts, line plots and scatter diagrams. Fits 16K. Tape of five routines and five demonstrations, including instructions . . . **£5** or SAE for details.
TopGraph Software
106 Blundell Road, Luton, Bedfordshire LU31 1SP.

SOFTWARE GAMES

SPECTRUM GOLF
The ORIGINAL Golf game for the Spectrum. Different course every game. Colour plan of each hole showing all obstacles. Scoring routines recognize birdies and eagles. Dramatic display for a hole in one.
"Interesting for golf fans" —
Sinclair User Feb. '83.
"Attractive and fun to play" —
Home Computing Weekly 5/4/83
Tee off today for **£3.95** Cassette (16/48K) with instructions from:
B.S. McALLEY, 1 COWLEAZE, CHINNOR, OXON.

SUPERB 3D strategy game, extremely challenging, machine coded (16/48K Spectrum), bargain £4.95. Supersoft, 91 Manor Road, London E17.

VALUE ZX-81 and Spectrum programs. Send SAE to MC Soft, 69 Sandfield Road, Oxford OX3 7RW today.

APY CASSETTE. Amazing cassette will allow you to stop, list or copy any previously unstoppable tape. Inclues two free utilities. State if for ZX81 or Spectrum. Send just £3. Bibker, 29 Chadderton Drive, Unsworth, Bury, Lancs.

WAR OF THE WORLDS
ZX Spectrum 16 or 48K
Save your miserable planet but beware of the dazzling stun rays! Features include progressive difficulty, high score, Hall of Fame and more.
Send a cheque or P.O. for **£4.95** to
OSCAR SYSTEMS, Dept ZX5, 13 Letchford Terrace, Harrow, Middx.
for this competitive and original game.

ZX81 16K MONOPOLY. Very addictive. Play the computer. Games last hours. Fast delivery cassette with instructions only £4.00. MPT, 42 Raedwald Drive, Bury St Edmunds, Suffolk.

SELEC SOFTWARE
1K ZX81
Machine Code Programs
Tape 1 Maze Guzzler/Super Breakout
Tape 2 The Invaders/Time Bomb
Tape 3 ZX-Panic/ZX-Squash
All programs 100% smooth, fast action machine code. The best 1K programs available.
MORE TITLES COMING SOON
ALL TAPES £3.50 (P&P FREE)
Send for free catalogue (SAE) or cheques/POs to:
SELEC SOFTWARE
37 Councillor Lane, Cheadle, Cheshire.

SERVICES

ZX REPAIR SERVICE
At last no need to wait for weeks
Send your faulty ZX81 for fast reliable repair. We will repair any ZX81 KIT or READY BUILT UNIT.
ONLY £15.95 fully guaranteed + **£1.75 p&p**
Please state the nature of problem. Send cheque or postal order to:
NEXT COMPUTER SYSTEMS
88 Harvest Road, Englefield Green, Surrey TW20 0QR.

T-SHIRTS. ZX81, Spectrum. White, blue, red, S/M/L/XL £3.25. One per cent screens. 12 Star Lane Estate, Great Wakering, Essex.

ACCESSORIES

*****ZX81 VIDSWITCH*****
INVERSE VIDEO MODULE
The Original and still the Best

Sharp white characters on completely black screen (no border). Module is built around an ultra-miniature toggle switch allowing switching between modes. Easy-build Kit **£3.95** Built **£4.95**.

NEW MINI KEYBOARD NEW

41 real moving keys which give audible feedback when pressed. Fits in recess left after peeling off ZX81 touch keyboard.
• Plug-in • No soldering • Faster programming • Two colour legends • No extra case needed • PCB finished in matt black •

KIT £22.50   BUILT £29.95
Repeat key similar to Spectrum £6.95
Two Tone keyboard bleeper £8.95

SPECTRUM BEEP BOOSTER

Plug-in **£6.95**

ZX81 16K SOFTWARE

Rescue **£4.95**
Frogs **£3.95**
Asteroids **£4.95**
Scramble **£4.95**
1K Games Pack (includes Invaders) **£4.95**

SPECTRUM SOFTWARE

Mad Martha (text and graphics) 48K .. **£5.95**
Scramble (Any Spectrum) **£5.50**
Cosmic Raiders (Any Spectrum) **£5.95**

Many others available. SAE for free catalogue. Cure white outs and top-line slant and display jitter with our Ram Pack stabiliser (Sinclair) and Heatsink Kit **£3.75**. All prices inclusive. Overseas orders welcome.

B. A. READER (Dept C)
45 Alfred Street, Kings Heath, Birmingham B14 7HG.

ZX81 VIDEO INVERTER saves your eyes, increases safe level, displays sharp white characters on solid black background. Screen kit £4, built £5 includes VAT and P&P instructions. Send cheque/postal order to D. Fritsch, 6 Standon Road, Thelwall, Warrington, Cheshire WA4 2HS.

ZX80/81 NANOS
"Quick Reference" Cards

Why waste time and frustration struggling with the books? This card has it all. "There is nothing like it in the world". Send **£3.50** per card to:

ELKAN ELECTRONICS (Dept ZX)
FREEPOST, 11 Bury New Road, Prestwich, Manchester M25 6LZ or Tel: 061-798 7613 (24 hours)

— other Sinclair items also available —

FOR ALL YOUR CLASSIFIED NEEDS CALL ASP LTD 01-437 1002

ADD-ONS

ZX81 OWNERS! Cure your Sinclair 16K Ram Pack wobble. Secure method of holding pack firmly. Easily added £4.50 (inc. p&p). Apply: F. Jones, 24 Highlands Drive, Burgess Hill, Sussex RH15 8JJ.

ZX Spectrum — CP48
32K Memory Extension

Gives total of 48K. Simply fits internally on series 2 models with no soldering. Similar to competitors but only **£23.95**

CITADEL PRODUCTS LTD
Dept. ZX, 50 High Street, Edgware, Middlesex HA8 7EP. Tel: 01-951 1848

ZX81 AND SPECTRUM SPEECH SYNTHESISERS

Completely self contained plug in modules giving unlimited vocabulary, using allophone speech synthesis. **£39.50**. Please state ZX81 or Spectrum. Dealer enquiries welcome.

SPIRIT INSTRUMENTS
Station Road, Maldon, Essex.

AGENCIES REQUIRED

DJS ENTERPRISES shortly opening retail Micro Computer Centre in South West requires additional hardware and software agencies. Box Bo ZX213, Argus Specialist Publications, 145 Charing Cross Road, London WC2 0EE.

WANTED

SPECTRUM high quality programs wanted urgently. Good royalties paid. Send cassettes, full listings and instructions to G-M-J Software, 29 Frenchgate, Richmond, N. Yorkshire.

APPOINTMENTS

GOOD ZX Spectrum or VIC Programmers (London based) to undertake paid projects in their own time. Phone 01-747 1373 (evenings).

SOFTWARE GAMES

SPECTRUM **dotty** 16KZX81
FROM
UNIQUE

Players must test their imagination by guessing the identity of a dotty picture as quickly as they can. Pictures are chosen at random from a wide selection, and a picture plotter is supplied, so the user can change pictures at will.
Dot by Dotty, all will be revealed

State machine and send cheques/PO's for only £3.40 to:-
UNIQUE COMPUTER SERVICES
Unit 6, 19 Silton Street
Manchester M9 1WS

RECRUITING?

Call ASP CLASSIFIED 01-437 1002

It's easy to complain about advertisements.

Every week, millions of advertisements appear in the press, on posters or in the cinema.

Most of them comply with the rules contained in the British Code of Advertising Practice and are legal, decent, honest and truthful.

But if you find one that, in your opinion, is wrong in some way, please write to us at the address below.

We'd like you to help us keep advertising up to standard.

The Advertising Standards Authority.
If an advertisement is wrong, we're here to put it right.

A.S.A. Ltd., Brook House, Torrington Place, London WC1E 7HN.

THE REAL INVESTMENT

NEW:
THE X-ROMCARD

FOR YOUR ZX81



WHAT IS *SO* SPECIAL ABOUT THE SPECIAL RAMPACK?



fig. 1



fig. 2

WHAT IS SO SPECIAL ABOUT AUDIO'S 16K SPECIAL RAM PACK?

THE SPECIAL RAM PACK (16K) is the latest design of its kind, following at least a dozen similar products. It has many of the good points of its predecessors, including packs manufactured by **Sinclair Research, Memotech, Bygbyte** and **Econotech** to cite a few of the best known.

With the experience we have gained with memories both for the **ZX 81** and a **wide range of other Micros**, we are also able to offer some **exclusive new features**:

THE CASE: **The SPECIAL RAM PACK** has been designed to overcome the infamous **'wobbling'** problem associated with many other packs, and does not resort to **temporary solutions** such as **Velcro, Blutak, adhesive strip** or a huge, unsightly case. It clamps positively to the ZX 81 by means of a special built in ridge aided by a **resilient spring effect** in the connection socket (see figure 1).

EXPANSIBILITY: **The SPECIAL RAM PACK** is not only compatible with all software and hardware produced for the **ZX 81**, such as the **ZX printer**, but it also has the edge over other packs with its **expansibility**. You will find it interesting that the **keyboard sounder** option is already included in the **SPECIAL RAM PACK**. It also has **expansion pins**, ready to receive — **in the same case** — the most advanced add-on ever designed for the ZX 81 — **the X ROM CARD** (see figure 2).

WHAT DOES THE X-ROM CARD OFFER?

1. **The X-ROM CARD has a built in autostart ROM.** Programs can be run automatically every time the ZX 81 is switched on. This will result in a huge increase factor for **ROM based software**, since all software houses are currently [illegible] aware of copyright piracy problems.
2. **The X-ROM CARD has a built in printer interface**, necessary if you wish to use [illegible] other than the XZ [illegible]. **This is invaluable for any serious applications**, including **word processing**, wh[illegible] presentation is of the utmost importance.
3. **The X-ROM CARD has a built in EPROM burner.** This means that you can save Basic or Machine code programs onto silicon chips and play them back at any time.
4. **You may of course buy the X-ROM CARD to use with other Ram Packs** such as the [illegible]lair **Ram Pack.** [illegible] when you purchase the X-ROM CARD with the SPECIAL RAM PACK, you will have the advantage of lower [illegible], greater reliability and neatness, since the whole system is nicely housed within a single case.

IN CONCLUSION, **The SPECIAL RAM PACK**, is the **best immediate investment** for your **ZX 81**. The availability of the **X-ROM CARD** opens the way to new software development such as **languages programming** and is the guarantee that your system will never be obsolete.

THE BEST GUARANTEE THAT YOUR ZX81 WON'T BE OBSOLETE FOR ONLY:

£ 19

including VA[illegible]

TECHNICAL SPECIFICATIONS:

- SPECIAL RAMPACK:
  - **Memory capacity:** 16K bytes.
  - **Maximum speed:** 200 ns access time.
  - **Power requirement:** Use exclusively Sinclair PSU.
- X-ROMCARD:
  - **Autostart ROM:** 4K byte, exchange for 8K bytes and software switch possible. Displays memory size, checks for byte "00" (identifies ZX basic) @ 2000H. Loads program if found, checks for byte "C3" and jumps (2000H) if found. Checks for presence of ROM in socket n.3 and ROM catalog, displays catalog if found. Contains also machine code monitor and printer utility.
  - **Preprogrammed ROM:** Catalog available on request. Use only 2732 or 2764 ROM/EPROM.
  - **Blank EPROM:** Use only 2764 — 8K bytes per device — in socket n.3. 3 × 9V, PP3 size batteries are needed to burn EPROM.
  - **Printer connection:** 16 pin DIL output, use standard IDC ribbon cable. Outputs include DO to D7, Strobe, Reset, Inputs include No-fault and Busy.
  - **Documentation:** Schematic diagram included, full listings of Autostart ROM extra (only for X-ROMCARD user, £1.50 + large SAE)

AUDIO-COMPUTERS

87 BOURNEMOUTH PARK ROAD,
SOUTHEND ON SEA – ESSEX SS5 2JJ